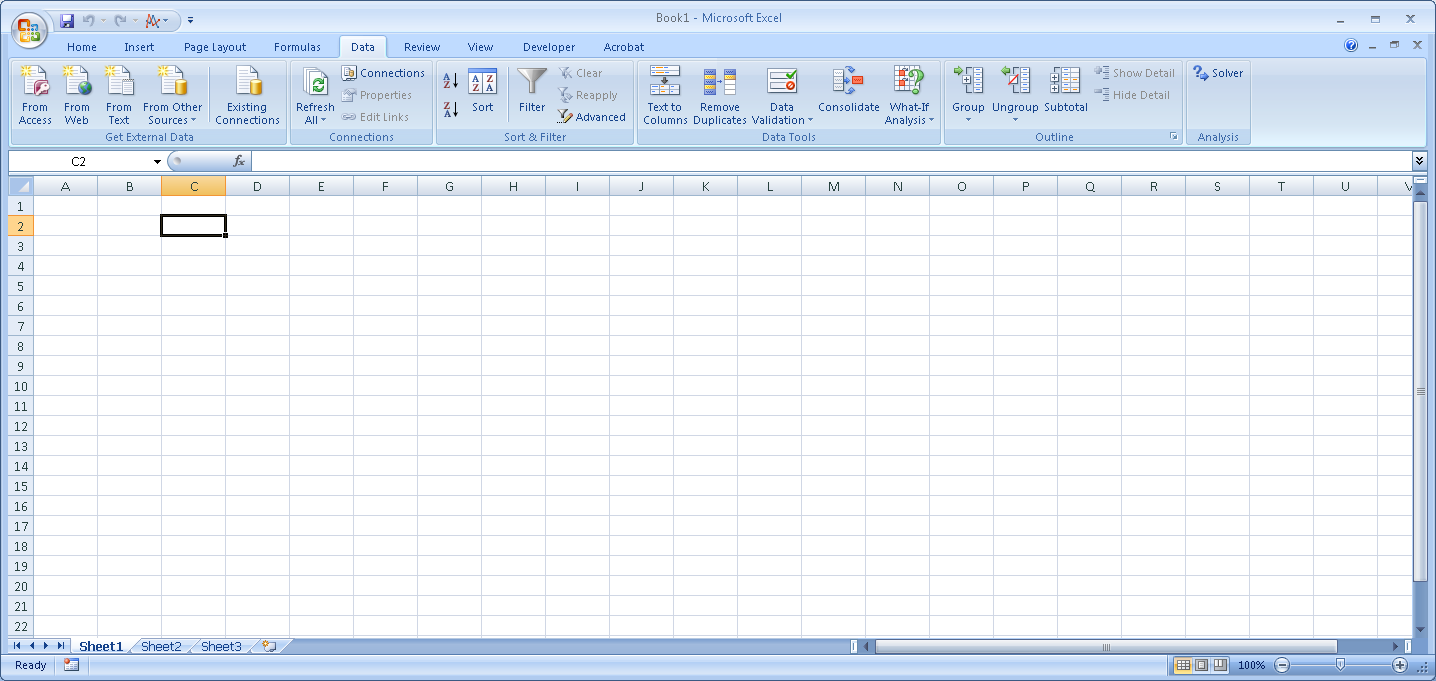
# CHAPTER 10

**INFORMATION SYSTEMS CONTROLS FOR SYSTEMS RELIABILITY – PART 3: PROCESSING INTEGRITY AND AVAILABILITY**

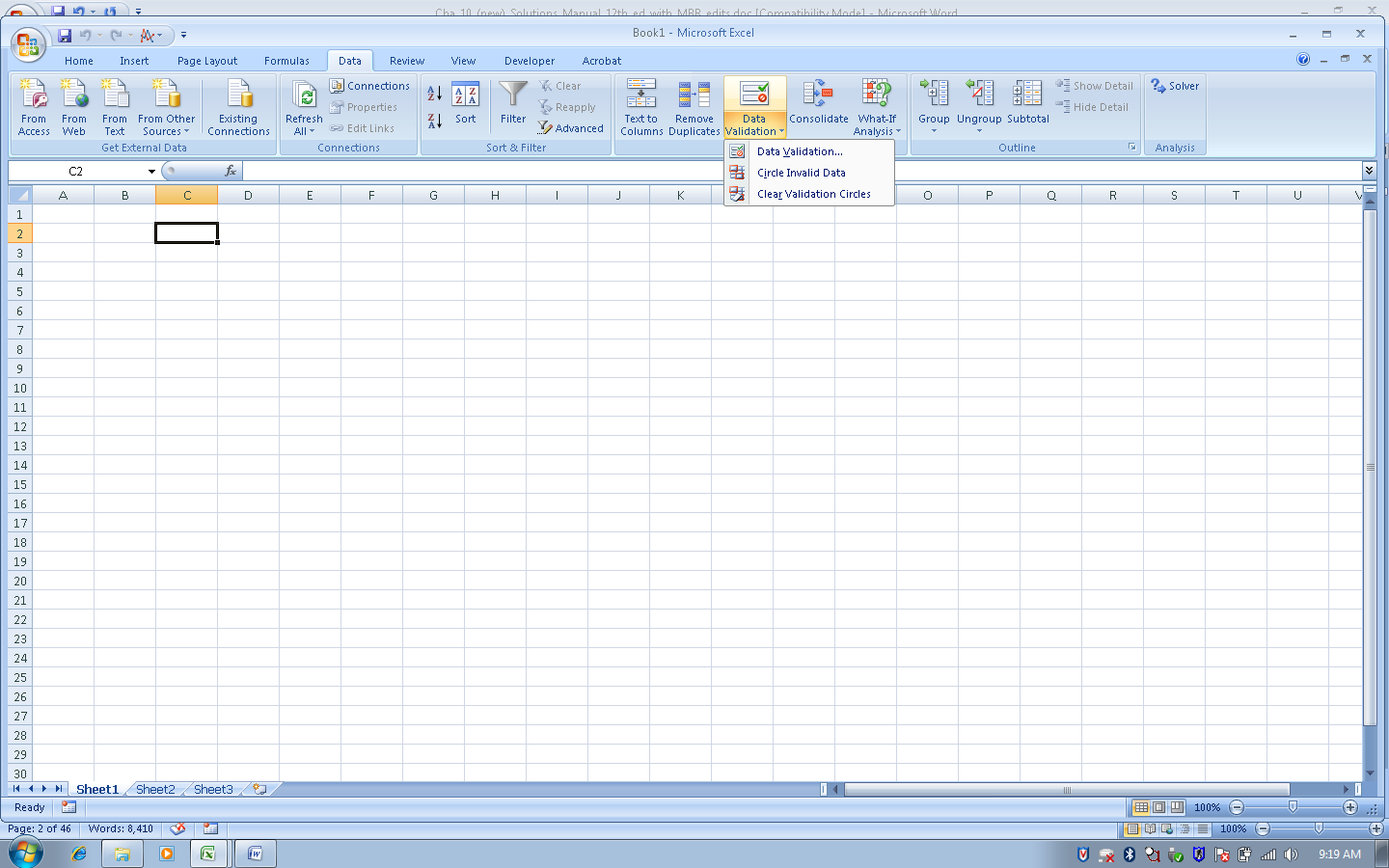
**SPECIAL INTRODUCTION TO EXCEL**

This chapter includes a number of problems that use Excel’s built-in Data Validation tool to help students better understand processing integrity controls by programming them in a spreadsheet. Some students will already be familiar with this tool, others will not. Therefore, this brief introductory tutorial may be useful as a hand-out prior to assigning the Excel questions in this chapter.

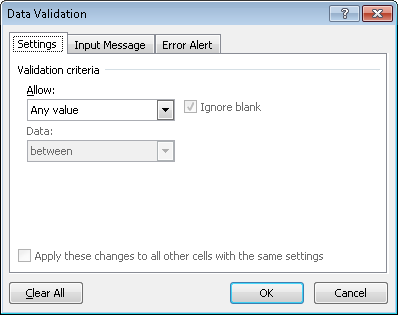
The Data Validation tool is found on the “Data” tab, as shown below:



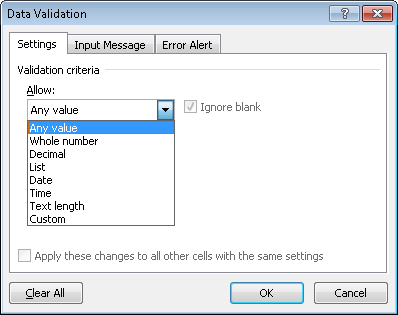
Click on “Data Validation” and then choose the option “data validation”:



This brings up the following window, which can be used to design a variety of processing integrity controls that will apply to the currently selected cell (in the example above, the Data Validation controls will be applied to cell C2):

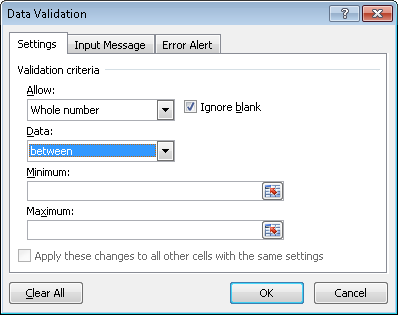


Clicking on the drop-down arrow in the “allow” box yields the following choices:



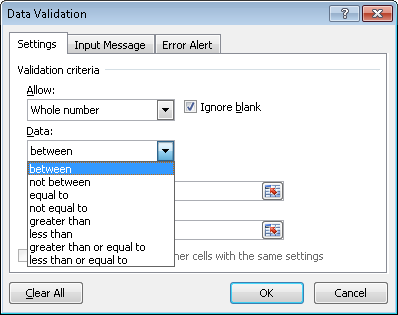
* Any value (the cell can take numeric, text, date, etc. input) without restrictions
* Whole numbers only allowed
* Decimals allowed (but not required)

Choosing either whole numbers or decimals, yields the following additional choices:



This default window can be used to create a “range check” with minimum and maximum values.

Click the drop-down arrow in the Data box to reveal other types of tests that can be created:

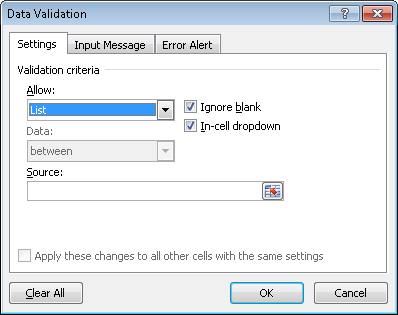


* List – permissible values must be selected from a list that the control designer creates

The list of permissible choices can appear in a drop-down menu (if that box is checked) using values found in a set of cells in the spreadsheet (using the source field):

If the “In-cell dropdown” box is checked, the values will appear in a drop-down list when a user clicks on that cell.

The list of permitted values in the drop-down box can be found in the portion of the spreadsheet as indicated in the “Source” box



If the “In-cell dropdown” box is not checked, users will still be restricted to entering values from the list indicated in the source box, but will have to manually type in those values rather than selecting from a drop-down menu.

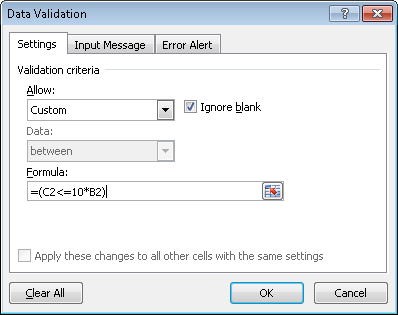
* Date – only date values
* Time – only time values
* Text Length – length of text string

Choosing either Date, Time, or Text Length yields the same set of choices as for “whole numbers” or “decimals”, making it easy to create limit checks, range checks, size checks, etc.:

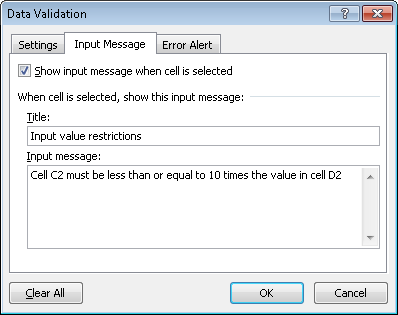


* Custom – formulas can be used to limit input values

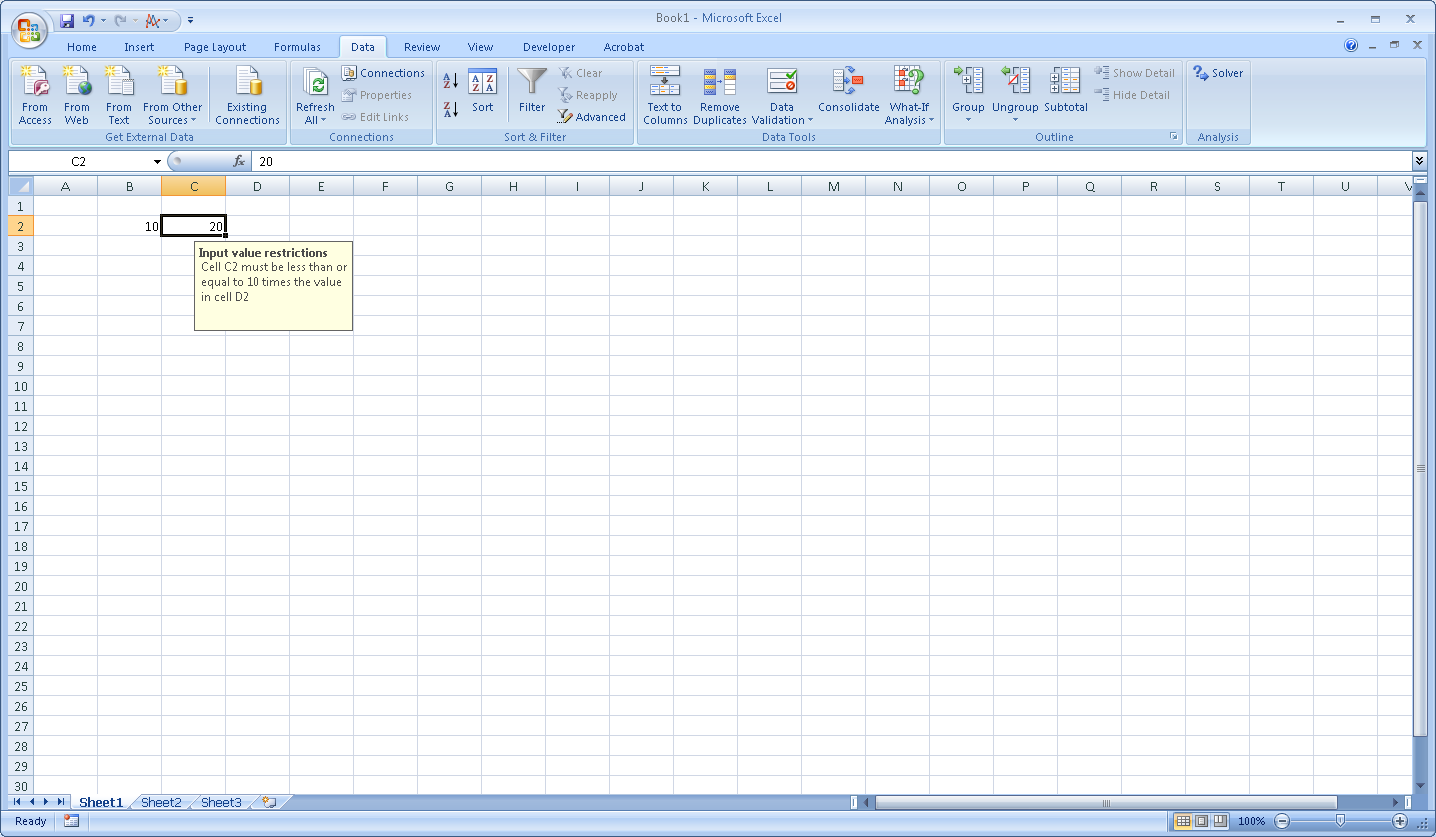
For example, we can create a “reasonableness test” that requires cell C2 to be less than or equal to 10 times the value in cell B2 as follows:



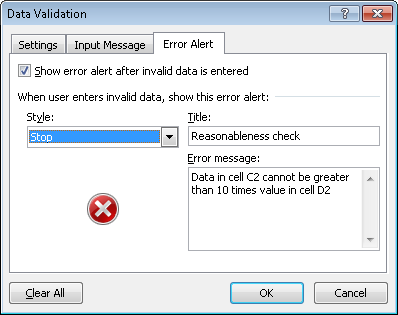
Once the processing integrity control has been designed, the “Input Message” tab can be used to create a message explaining the permissible input values that will appear whenever a user selects that cell:



Which yields the following:

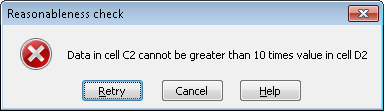


Finally, the “Error Alert” tab can be used to create a meaningful error message whenever user data violates the constraints:

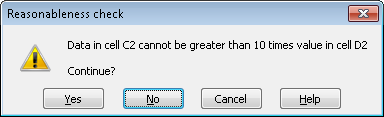


The message can have a title, plus as much text as desired. In addition, there are three action choices:

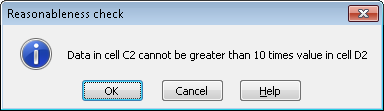
1. Stop – the user is prohibited from inputting the erroneous data



1. Warning – the user is informed that the data is not valid, but has the option of entering it anyway.



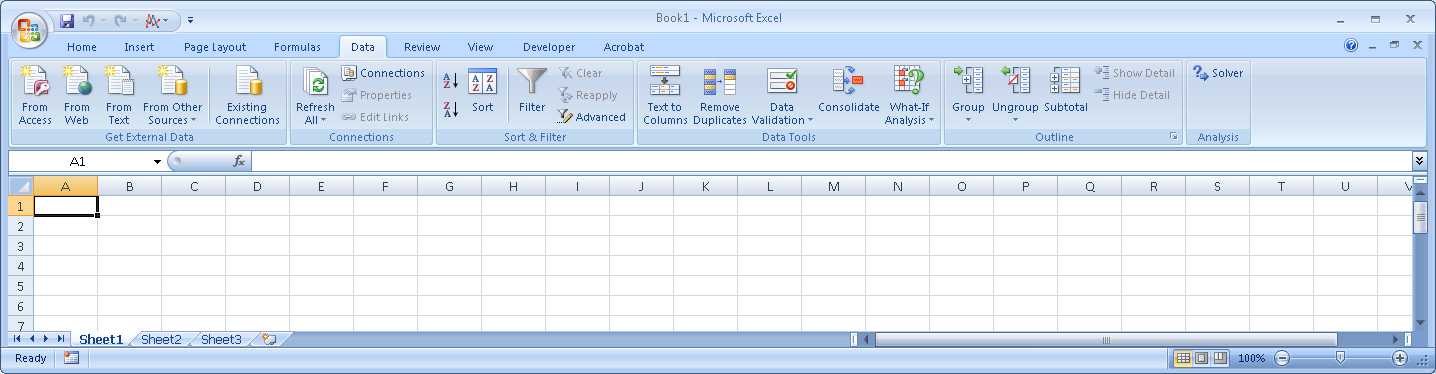
1. Information – the user is informed that the data is not valid. Clicking OK results in the data being entered anyway; clicking cancel rejects the data.



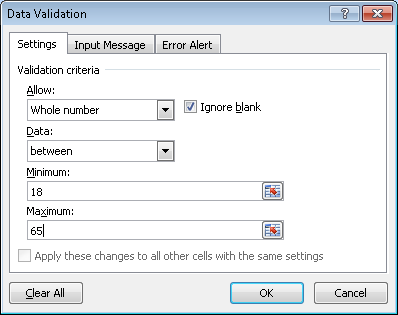
**SUGGESTED ANSWERS TO DISCUSSION QUESTIONS**

**10.1** **Two ways to create processing integrity controls in Excel spreadsheets are to use the built-in Data Validation tool or to write custom code with IF statements. What are the relative advantages and disadvantages of these two approaches?**

Excel provides a “Data Validation” tool on the Data tab:



The Data Validation tool serves as a “wizard” to program a variety of input editing/ processing controls. For example, if you want to limit the values in cell A1 to be between 18 and 65, you could use the Data Validation tool to program this range check as follows:



The “Input Message” tab can be used to inform the user what values are permissible. The “Error Alert” tab can be used to create an error message that will be displayed if the values are not permissible (in the case of this example, if the values are either less than 18 or greater than 65).

The same range check could be programmed using an IF statement, as follows:

=IF(AND(A1>=18,A1<=65),"","Error: values must be between 18 and 65")

An IF statement consists of three arguments, separated by commas: =IF(first argument, second argument, third argument). The first argument is the test to be performed, the second controls what happens if the test is true, and the third argument controls what happens if the test is false. In this example, the first argument is testing whether the value in cell A1 is between 18 and 65, inclusive. The second argument directs that if the test is true, no error message should be displayed (the two double-quote marks indicate that nothing will be displayed). The third argument controls what happens if the test is not true. In this example, if the value entered into cell A1 is less than 18 or greater than 65, the message “Error: values must be between 18 and 65” will be displayed.

The Data Validation tool is easier to use. However, it is limited to performing tests of just one condition. More complex tests require the IF function. For example, perhaps we want to treat values of 18, 19, and 20 different from values 21-65. This can be done by nesting IF statements, as follows:

=IF(A1>=18,IF(A1<21,"value is 18-20",IF(A1<=65,"value is between 21 and 65","Error: value must be less than or equal to 65")),"Error: Value must be greater than or equal to 18")

This formula works as follows:

Step 1: the first IF statement tests whether the value in cell A1 is greater than or equal to 18. If it is true, then it proceeds to evaluate the second if statement. If the value entered is less than 18, it returns the final error message: “Value must be greater than or equal to 18”

Step 2: If the first IF statement is true (i.e., the value in cell A1 is greater than or equal to 18) the next test is whether the value is less than 21. If it is, then the message “value is 18-20” is displayed. If the value in A1 is greater than or equal to 21, a third test is performed, testing whether it is less than or equal to 65.

Writing IF statements requires careful thought, but provides total flexibility in creating very complicated processing integrity checks.

**10.2** **What is the difference between using check digit verification and a validity check to test the accuracy of an account number entered on a transaction record?**

Check digit verification is designed to detect typographical errors such as transposing two digits or entering the wrong digit (e.g., typing an 8 instead of a 3). Passing a check digit verification test only ensures that the account number could exist.

A validity check verifies that the account number actually does exist, by searching for it in a master file. Check digit verification can be done at the point of data entry; a validity test requires accessing the relevant master file and takes time to search the account number field in that file to see if it contains a specific value.

**10.3** **For each of the three basic options for replacing IT infrastructure (cold sites, hot sites, and real-time mirroring) give an example of an organization that could use that approach as part of its DRP. Be prepared to defend your answer.**

Many solutions are possible. The important point is to justify that the method yields an appropriate RTO for the organization. Cold sites yield RTOs measured in days; hot sites result in RTOs measured in hours; and real-time mirroring have RTOs measured in minutes. Here are some possible examples:

Cold site: smaller businesses, such as a local CPA firm. In most situations, CPA firms can probably function without their main information system for a day or a couple of days. Most employees have laptops and could continue to do much of their work (collecting audit evidence, writing reports, working on spreadsheets) and then upload their work to the main servers once the cold site is up and running.

Hot site: Many businesses could function for several hours using paper-based forms until their data center was back up and running. For example, if a retailer’s information system went down, new sales orders could be processed on paper and entered later.

Real-time mirroring: Internet-only companies need this because they can only earn revenue when their web site is up and running. Nor can airlines and financial institutions operate using paper-based forms; they need to have a backup system available at all times.

**10.4 Use the numbers 10–19 to show why transposition errors are always divisible by 9.**

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | B - A | Divisible by 9? |
| Original Number | Transposed Number | Difference |  |
| 10 | 01 | 9 | Yes |
| 11 | 11 | 0 | Not a transposition |
| 12 | 21 | 9 | Yes |
| 13 | 31 | 18 | Yes |
| 14 | 41 | 27 | Yes |
| 15 | 51 | 36 | Yes |
| 16 | 61 | 45 | Yes |
| 17 | 71 | 54 | Yes |
| 18 | 81 | 63 | Yes |
| 19 | 91 | 72 | Yes |

When two numbers are transposed, the difference between the original number and the transposed number is divisible by 9 except when the two digits have the same value.

**10.5** **What are some business processes for which an organization might use batch processing?**

Batch processing may be used when master files do not need to be updated in real-time. For example, many organizations process accounts payable in batches once a day or once a week because they do not need up-to-the-minute accuracy about the balances they owe to suppliers. In contrast, accounts receivable benefits from on-line processing because organizations need to know whether a new order will exceed a customer’s credit limit. Batch processing is also appropriate for business processes such as payroll and dividend payments that only happen periodically but affect virtually every account in a master file.

**10.6**  **Why do you think that surveys continue to find that a sizable percentage of organizations either do not have formal disaster recovery and business continuity plans or have not tested and revised those plans for more than a year?**

Likely reasons include:

* Belief that “it won’t happen to us”
* Lack of time to develop plans
* Lack of money to develop plans
* Not important to senior management (no support for planning or testing)
* Risk attitude/appetite of senior management

**SUGGESTED SOLUTIONS TO THE PROBLEMS**

**10.1 Match the following terms with their definitions:**

|  |  |
| --- | --- |
| **\_\_s\_\_ 1. business continuity plan (BCP)** | 1. **A file used to store information for long periods of time.** |
| **\_\_j\_\_ 2. completeness check** | 1. **A plan that describes how to resume IT functionality after a disaster.** |
| **\_\_o\_\_ 3. hash total** | 1. **An application control that verifies that the quantity ordered is greater than 0.** |
| **\_\_u\_\_ 4. incremental daily backup** | 1. **A control that verifies that all data was transmitted correctly by counting the number of odd or even bits.** |
| **\_\_a\_\_ 5. archive** | 1. **An application control that tests whether a customer is 18 or older.** |
| **\_\_v\_\_ 6. field check** | 1. **A daily backup plan that copies all changes since the last full backup.** |
| **\_\_c\_\_ 7. sign check** | 1. **A disaster recovery plan that contracts for use of an alternate site that has all necessary computing and network equipment, plus Internet connectivity.** |
| **\_\_w\_\_ 8. change control** | 1. **A disaster recovery plan that contracts for use of another company’s information system.** |
| **\_\_i\_\_ 9. cold site** | 1. **A disaster recovery plan that contracts for use of an alternate site that is pre-wired for Internet connectivity but has no computing or network equipment.** |
| **\_\_e\_\_ 10. limit check** | 1. **An application control that ensures that a customer’s ship-to address is entered in a sales order.** |
| **\_\_k\_\_ 11. zero-balance test** | 1. **An application control that makes sure an account does not have a balance after processing.** |
| **\_\_n\_\_ 12. recovery point objective (RPO)** | 1. **An application control that compares the sum of a set of columns to the sum of a set of rows.** |
| **\_\_m\_\_ 13. recovery time objective (RTO)** | 1. **A measure of the length of time that an organization is willing to function without its information system.** |
| **\_\_p\_\_ 14. record count** | 1. **The amount of data an organization is willing to re-enter or possibly lose in the event of a disaster.** |
| **\_\_r\_\_ 15. validity check** | 1. **A batch total that does not have any intrinsic meaning.** |
| **\_\_t\_\_ 16. check digit verification** | 1. **A batch total that represents the number of transactions processed.** |
| **\_\_x\_\_ 17. closed-loop verification** | 1. **An application control that validates the correctness of one data item in a transaction record by comparing it to the value of another data item in that transaction record.** |
| **\_\_d\_\_ 18. parity checking** | 1. **An application control that verifies that an account number entered in a transaction record matches an account number in the related master file.** |
| **\_\_q\_\_ 19. reasonableness test** | 1. **A plan that describes how to resume business operations after a major calamity, like Hurricane Katrina, that destroys not only an organization’s data center but also its headquarters.** |
| **\_\_y\_\_ 20. financial total** | 1. **A data-entry application control that verifies the accuracy of an account number by recalculating the last number as a function of the preceding numbers.** |
| **\_\_z\_\_ 21. turnaround document** | 1. **A daily backup procedure that copies only the activity that occurred on that particular day.** |
|  | 1. **A data-entry application control that could be used to verify that only numeric data is entered into a field.** |
|  | 1. **A plan to ensure that modifications to an information system do not reduce its security.** |
|  | 1. **A data-entry application control that displays the value of a data item and asks the user to verify that the system has accessed the correct record.** |
|  | 1. **A batch total that represents the total dollar value of a set of transactions.** |
|  | 1. **A document sent to an external party and subsequently returned so that preprinted data can be scanned rather than manually reentered.** |

**10.2 Excel Problem**

**Enter the following data into a spreadsheet and then perform the following tasks:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Employee Number** | **Pay rate** | **Hours worked** | **Gross Pay** | **Deductions** | **Net pay** |
| **12355** | **10.55** | **38** | **400.90** | **125.00** | **275.90** |
| **2178g** | **11.00** | **40** | **440.00** | **395.00** | **45.00** |
| **24456** | **95.00** | **90** | **8550.00** | **145.00** | **8405.00** |
| **34567** | **10.00** | **40** | **400.00** | **105.00** | **505.00** |

1. **Calculate examples of these batch totals:**
   * **A hash total**

Solution: sum of the employee number or pay rate columns, since these totals have no intrinsic meaning. In this example, the error in the second employee’s number would prevent calculating a hash total on that column. So you could only sum the pay rate column, yielding a hash total of 126.55

* + **A financial total**

Solution: sum of the hours worked (208), gross pay (9790.90), deductions (770), or net pay (9,230.90) columns as all these results have financial meaning

* + **A record count**

Solution: 4, which is a count of the rows

1. **Assume the following rules govern normal data:**
   * **Employee numbers are five-digits in length and range from 10000 through 99999.**
   * **Maximum pay rate is $25, and minimum is $9.**
   * **Hours worked should never exceed 40.**
   * **Deductions should never exceed 40% of gross pay.**

**Give a specific example of an error or probable error in the data set that each of the following controls would detect:**

* + **Field check**

A field check on the employee number column would detect that the second row does not contain only numbers; thus, it would detect the letter “g” in the employee number.

* + **Limit check**

A limit check on pay rate could flag row 3 as an error because $95 exceeds the maximum pay rate of $25,)

A limit check on hours worked would also flag row 3 as an error because 90 hours worked exceeds the maximum hours worked of 40.

* + **Reasonableness test**

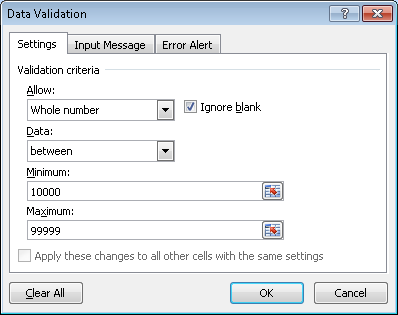
Comparison of deductions to gross pay would flag a potential problem in row 2 since it is not unlikely that a person being paid $440 have $395 of deductions.

* + **Cross-footing balance test**

A cross-footing balance test would detect that sum of gross pay (9790.90) minus sum of deductions (770) does not equal sum of net pay (9230.90).

1. **Create a control procedure that would prevent, or at least detect, each of the errors in the data set.**
   * **Employee number not numeric**

Using the data validation tool, select the cells you want to test (in the employee number column) and specify the legal limits (whole numbers beginning with 10000 through 99999) as follows:



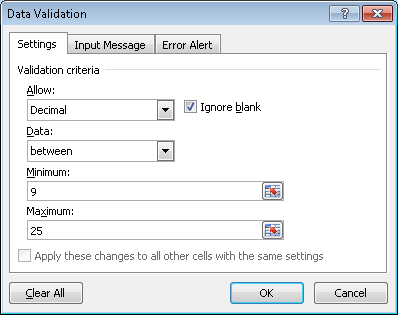
Alternatively, you could write the following IF statement to perform the same test:

=IF(AND(E3>=10000,E3<=99999),"","error")

This tests whether the cell value is both greater than or equal to 10000 and also less than or equal to 99999. If it is, nothing will be displayed (the two double quote marks tell Excel to not display any error message). If the value of the cell falls outside the allowed limits, an error message is displayed.

* + **Pay rate too high or too low**

This range test could be programmed using the data validation tool as follows:



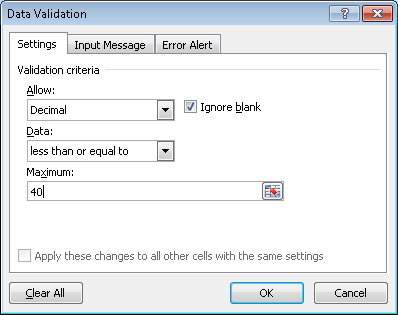
Alternatively, this logical test would catch such errors and display an appropriate error message:

=IF(D6<9,"pay rate must be at least $9",IF(D6>25,"pay rate must be less than $25",""))

This formula first tests whether the pay rate in the cell is less than the minimum allowable rate of $9. If this test is true, an error message is displayed that specifically states that the pay rate must be at least $9. If the test if false, then a second IF statement is evaluated to check whether the pay rate in the cell is greater than the maximum allowable rate of $25. If it is, the appropriate error message is displayed. If the second IF statement is false, then it means that the pay rate in the cell must be between $9 and $25 (because the second IF test is only evaluated if the first one is true), so no error message is displayed (hence the two double-quotes).

* + **Hours worked too high**

Using the data validation tool, a limit check to ensure that hours worked must be less than or equal to 40 can be designed as follows:



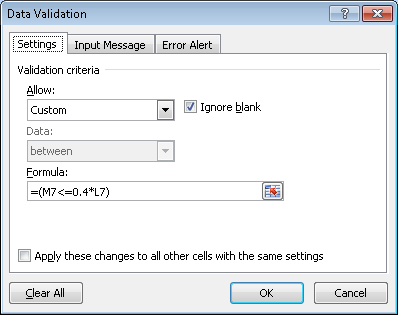
Alternatively, the following IF statement would enforce the same limit check:

=IF(A4<=40,””,”Error: hours worked cannot exceed 40”)

The IF test checks whether the value in cell A4 is less than or equal to 40. If it is, then no error message is displayed (the two double-quotes say to display nothing). If the value in cell A4 is greater than 40, the test fails and the error message is displayed.

* + **Deductions too high relative to gross pay**

This reasonableness test would be programmed using the data validation tool and choosing “custom” in the allow field, as follows:



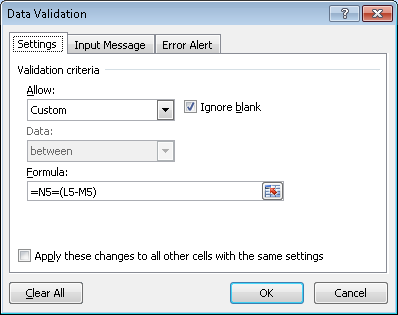
The formula would limit the deductions in cell M7 to be less than or equal to 40% of the gross pay in cell L7.

Alternatively, the following IF statement would perform the same reasonableness test:

=IF(M7/L7<=0.4,””,"deductions exceed 40% of gross pay")

The IF statement would test whether the deductions in cell M7 are less than or equal to 40% of gross pay in cell L7. If the test is true, no error message would be displayed (the two double-quotes for the second argument of the IF formula). If the test is false, the error message in the third argument of the IF formula would be displayed.

* + **Error in calculating net pay**



Alternatively, the following IF statement would catch the error:

=IF(L5-M5=N5,"","net pay does not equal gross pay - deductions")

**10.3 Excel Problem**

**The Moose Wings Cooperative Flight Club owns a number of airplanes and gliders. It serves fewer than 2,000 members, who are numbered sequentially from the founder, Tom Eagle (0001), to the newest member, Jacques Noveau (1368). Members rent the flying machines by the hour, and all must be returned on the same day. The following six records were among those entered for the flights taken on September 1, 2010:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Member #** | **Flight Date**  **MM/DD/YY** | **Plane Used** | **Takeoff time** | **Landing time** |
| **1234** | **09/10/10** | **G** | **6:25** | **8:46** |
| **4111** | **09/01/10** | **C** | **8:49** | **10:23** |
| **1210** | **09/01/10** | **P** | **3:42** | **5:42** |
| **0023** | **09/01/10** | **X** | **1:59** | **12:43** |
| **012A** | **09/01/10** | **P** | **12:29** | **15:32** |
| **0999** | **09/01/10** | **L** | **15:31** | **13:45** |

**Valid plane codes (plane used column): C = Cessna, G = glider, L = Lear Jet, P = Piper Cub)**

1. **Identify and describe any errors in the data.**

Five of the six records contain errors as follows:

1st - Wrong date is used (September 10 instead of September 1).

2nd - Member number is outside range (4111 is greater than 1368).

4th - Plane code X is not valid.

5th - Member number contains a character (A).

6th - Plane landing time (13:45) is earlier than the take off time (15:31).

1. **For each of the five data fields, suggest one or more input edit controls that could be used to detect input errors.**

Field 1 - Member number:

* Range check to verify that the field contains only four digits within the range of 0001 to 1368.
* Validity check on member number if a file of valid member numbers is maintained.

Field 2 - Date of flight start:

* Check that day, month, and year corresponds to the current date.
* Field check that value is a date

Field 3 - Plane used:

* Validity check that character is one of the legal characters to describe a plane (G, C, P, or L).
* Field check to verify that only a single character is used.)

Field 4 - Time of take off:

* Field check to verify that the field contains valid time format.

Field 5 - Time of landing:

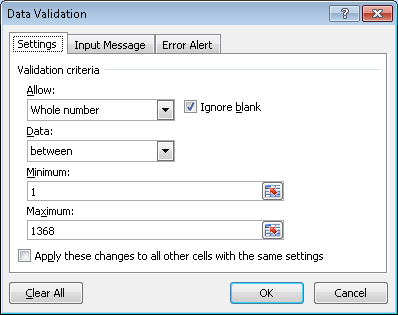
* Field check to verify that the field contains valid time format.
* Reasonableness test that field 5 is greater than field 4.

1. **Enter the data in a spreadsheet and create appropriate controls to prevent or at least detect the input errors.**

Field 1 - Member number:

* Range check to verify that the field contains only four digits within the range of 0001 to 1368.

Using the Data Validation tool in Excel (under the Data tab) this range check could be programmed as follows:



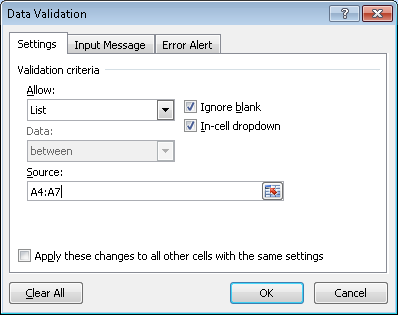
Alternatively, the following IF statement would do the same thing:

=IF(AND(A4>0,A4<1369),””,”Error: Values must be between 1 and 1368”)

The first argument tests whether the cell value for member numbers is a whole number that is greater than 0 and less than 1369 (you could also code this as greater than or equal to 1 and less than or equal to 1368). If the test is true, no error message is displayed (the two double quotes in argument 2 of the IF function). If the test is false, the error message displayed in the quotes in the third argument is displayed.

* + Validity check on member number if a file of valid member numbers is maintained.

Using the data validation tool, the validity check would be programmed as follows:



This tools says that the value input must match a list of legal values that are found in cells A4:A7 (which would hold the values C, G, L and P)

Alternatively, the following IF statement would perform the same test:

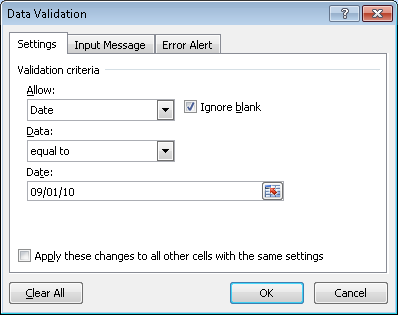
=IF(OR(G8=”C”,G8=”G”,G8=”L”,G8=”P”),””,”Error: Invalid plane code”)

The OR test checks the value of cell G8 against the four permissible values. If any match, the test is true and nothing is displayed. If none of the four tests matches, then the error message in the third argument is displayed.

Field 2 - Date of flight start:

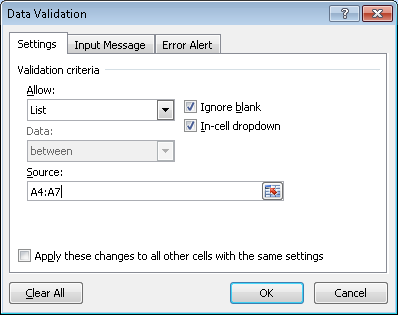
* Check that day, month, and year correspond to the current date.

In the data validation tool, you would select the cells you want to test and enter the date value you want to compare to, as follows:



Field 3 - Plane used:

* Validity check that character is one of the legal characters to describe a plane (G, C, P, or L).



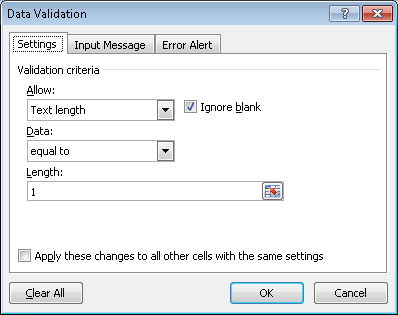
This tools says that the value input must match a list of legal values that are found in cells A4:A7 (which would hold the values C, G, L and P)

Alternatively, the following IF statement would perform the same test:

=IF(OR(G8=”C”,G8=”G”,G8=”L”,G8=”P”),””,”Error: Invalid plane code”)

The OR test checks the value of cell G8 against the four permissible values. If any match, the test is true and nothing is displayed. If none of the four tests matches, then the error message in the third argument is displayed.

* Check that only a single character is used. (field check)



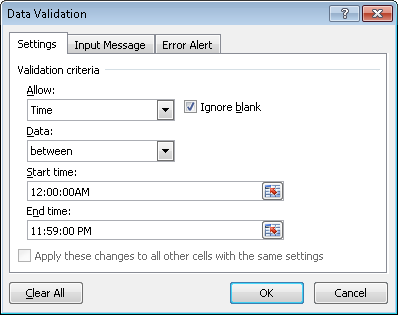
Alternatively, the following IF statement also checks this:

=IF(LEN(S4)=1,"","Plane character must contain only one character")

The LEN function returns the length of a text string. In this case, it checks the cell containing the plane code to verify that it is only 1 letter. If the test is true, no error message is displayed (the second argument of the IF statement has two double-quotes). If the test is false, it displays the error message in the third argument of the IF function.

Field 4 - Time of take off:

* Field check to verify that the field contains valid time format.

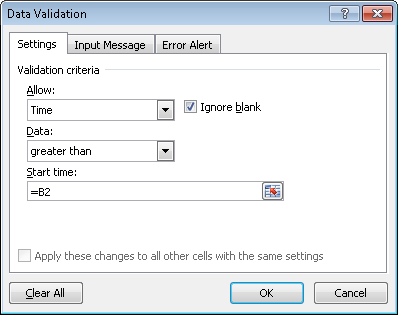


Field 5 - Time of landing:

* Field check to verify that the field contains valid time format.

Same as for field 4

* Reasonableness test that field 5 is greater than field 4.



1. **Suggest other controls to minimize the risk of input errors**.

* prompting to request each required input item.
* preformatting to display an input form including all required input items.
* completeness check on each input record to ensure all item have been entered.
* default values such as today’s date for the flight date.
* closed-loop verification (member name would appear immediately after the member number)

(SMAC Examination, adapted)

**10.4 The first column in Table 10-3 lists transaction amounts that have been summed to obtain a batch total. Assume that all data in the first column are correct. Cases a through d each contain an input error in one record, along with a batch total computed from that set of records.**

**For each case (a-d), compute the difference between the correct and erroneous batch totals and explain how this difference could help identify the cause of the error.**

Solution: Differences between the correct transactions column and the batch totals obtained after processing (Case A through D columns):

(a) (b) (c) (d)

$57,607.24 $57,607.24 $57,607.24 $57,607.24

- 57,616.24 -51,607.24 -48,807.24 - 56,952.92

($ 9.00) $ 6,000.00 $8,800.00 $ 654.32

Analysis of these differences:

1. The difference of $9 is evenly divisible by 9, which suggests the possible transposition of adjoining digits in the hundredths and tenths columns. More careful inspection indicates that the amount $1,978.95 from the correct transactions calculation was incorrectly transposed to $1,987.95 in the Case A calculation.
2. A difference of $6,000 represents a discrepancy in only one column, the thousands column. A possible error in transcribing one digit in that column is indicated. More careful examination reveals that the amount $7,832.44 from the correct transactions column was incorrectly recorded as $1,832.44 in the Case B column.
3. The difference of $8,800.00 is not divisible evenly by 9, which rules out a transposition error. The difference affects multiple columns, which rules out a single transcription error. The difference amount is not equal to any of the entries in the correct transactions batch total calculation, which rules out an error of omission. Dividing the difference by 2 gives $4,400.00, which is one of the entries in the correct transactions column. More careful inspection reveals that this amount has been inadvertently subtracted from the Case C batch total calculation rather than added.
4. The difference of $654.32 is not divisible evenly by 9. However, this amount is equal to one of the entries in the correct transactions column. Inspection reveals that this item was inadvertently omitted from the Case D column.

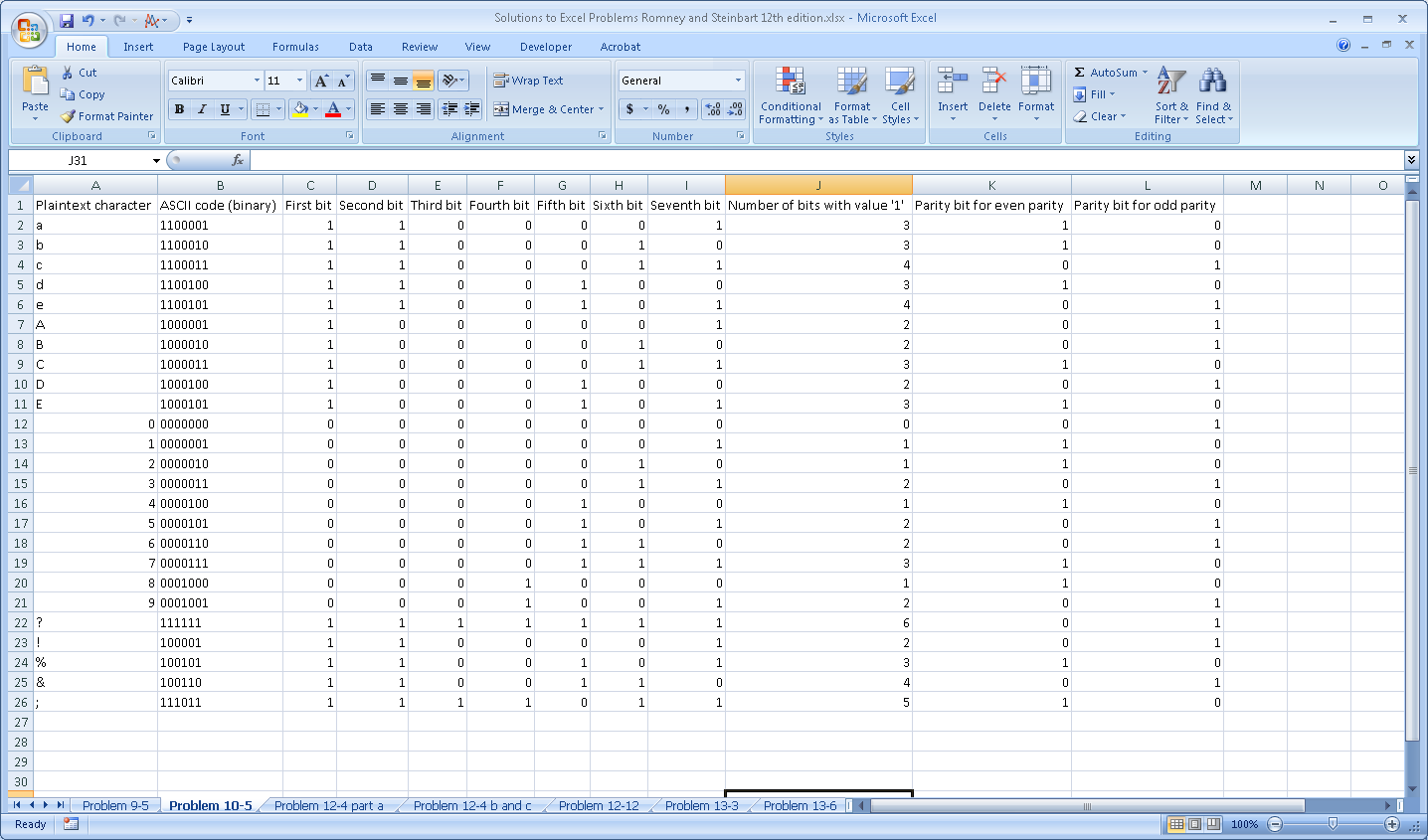
**10.5 Excel Problem**

**Create a spreadsheet with the following columns:**

* **Plaintext character**
* **ASCII code (7-bits, binary number)**
* **First bit**
* **Second bit**
* **Third bit**
* **Fourth bit**
* **Fifth bit**
* **Sixth bit**
* **Seventh bit**
* **Number of bits with value = 1**
* **Parity bit for odd parity coding**
* **Parity bit for even parity coding**

1. **Enter the 26 letters a-z (lowercase) and the ten digits (0-9) in the plaintext column**
2. **The ASCII column should convert the plaintext character to the binary code used by your computer.**
3. **The next seven columns should each display one bit of the ASCII code, beginning with the leftmost digit. (Hint: Excel provides text functions that can select individual characters from a string).**
4. **The tenth column should sum the number of bits that have the value ‘1’. (Hint: the text functions used to populate columns 3-9 return a text string that you will need to convert to a numeric value).**
5. **The eleventh column should have a 1 if the number in the tenth column is odd and 0 if the number in the tenth column is even.**
6. **The twelfth column should have a 1 if the number in the tenth column is even and a 0 if the number in the tenth column is odd.**

The solution should look like this:

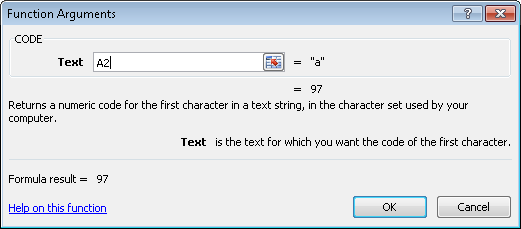


NOTE: Tell students that one of the objectives of this exercise (besides illustrating how parity bits work) is for them to explore the large number of built-in Excel functions. You may wish to provide one or two examples from the solution to get them started.

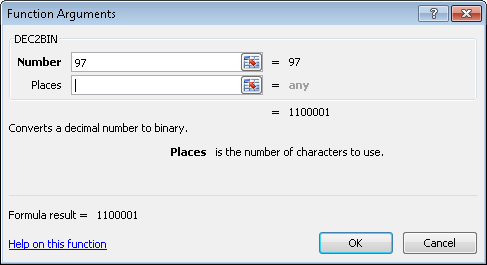
Functions used to populate columns in the solution:

Column b: converting the ASCII character in column A to its binary equivalent. This is accomplished by using the DEC2BIN and CODE functions: =DEC2BIN(CODE(A2))

* The CODE function is one of Excel’s built-in Text functions. It takes one argument, which in this case is the reference to the cell that contains the plaintext (cell A2), and returns the computer’s code set. The result for the lowercase letter “a” is 97:



* The DEC2BIN function is one of Excel’s built-in Engineering functions. It transforms a number, in this case the result of the CODE function, into binary (0s and 1s):



Columns C-I: the individual bits in the binary string. These are found using Excel’s Text functions as follows:

* Column C: =VALUE(LEFT(B2))

The LEFT function with only one argument returns a string representing the left-most digit in the reference cell. In this case, the reference cell (B2) contains the binary representation of the letter “a” = 1100001. Thus, the LEFT function returns a text string of 1.

The VALUE function converts a text value into a number. In this case, it converts the text of “1” into the number 1. This is necessary for the step of counting the number of bits with a value of 1 in order to calculate the parity bit (column J).

* Column D: =VALUE(LEFT(RIGHT(B2,6))). The combination of LEFT and RIGHT functions is used to return the second digit from the left in the binary number 1100001. The RIGHT function can take two arguments: the cell containing the numeric value to be manipulated (in this case B2) and the number of digits, beginning with the rightmost one, to return. In this case, it returns the 6 right-most digits: 100001. Next, the LEFT function lops off the left-most digit in that string, yielding text string of “1”. Finally, the VALUE function converts that text into the number 1.
* Column E: =VALUE(LEFT(RIGHT(B2,5))). The combination of LEFT and RIGHT functions is again used to return the third digit from the left in the binary number 1100001. The RIGHT function can take two arguments: the cell containing the numeric value to be manipulated (in this case B2) and the number of digits, beginning with the rightmost one, to return. In this case, it returns the 5 right-most digits: 00001. Next, the LEFT function lops off the left-most digit in that string, yielding text string of “0”. Finally, the VALUE function converts that text into the number 0.
* Column F: =VALUE(LEFT(RIGHT(B2,4))). The combination of LEFT and RIGHT functions is used to return the fourth digit in the binary number 1100001. The RIGHT function can take two arguments: the cell containing the numeric value to be manipulated (in this case B2) and the number of digits, beginning with the rightmost one, to return. In this case, it returns the 4 right-most digits: 0001. Next, the LEFT function lops off the left-most digit in that string, yielding text string of “0”. Finally, the VALUE function converts that text into the number 0.
* Column G: =VALUE(LEFT(RIGHT(B2,3))). The combination of LEFT and RIGHT functions is used to return the fifth digit in the binary number 1100001. The RIGHT function can take two arguments: the cell containing the numeric value to be manipulated (in this case B2) and the number of digits, beginning with the rightmost one, to return. In this case, it returns the 3 right-most digits: 001. Next, the LEFT function lops off the left-most digit in that string, yielding text string of “0”. Finally, the VALUE function converts that text into the number 0.
* Column H: =VALUE(LEFT(RIGHT(B2,2))). The combination of LEFT and RIGHT functions is used to return the sixth digit in the binary number 1100001. The RIGHT function can take two arguments: the cell containing the numeric value to be manipulated (in this case B2) and the number of digits, beginning with the rightmost one, to return. In this case, it returns the 4 right-most digits: 01. Next, the LEFT function lops off the left-most digit in that string, yielding text string of “0”. Finally, the VALUE function converts that text into the number 0.
* Column I: =VALUE(RIGHT(B2)). The RIGHT function with just one argument is used to return the right-most digit in the reference cell (B2), which in this case is the binary number 1100001. It returns a text string of “1”. The VALUE function then converts that text into the number 1.

COLUMN J: the number of bits with the value 1. Since columns C through I contain either the number 1 or the number 0, a simple SUM(C:I) yields the number of bits with the value of 1.

COLUMN K: Calculate the parity bit if using even parity. Even parity means that there should be an even number of bits, including the parity bit, that have a value of 1. Therefore, if the value in column J is odd (there are an odd number of bits in the 7-digit binary number representation of the plaintext character in that row) then the parity bit must be set to 1 in order to yield an even number of bits with the value 1. For example, in row 2, the binary representation of the lowercase letter “a” is 1100001 which, as shown in Column J, contains an odd number of bits with a value of 1. Therefore, the parity bit for the lowercase letter “a” must be set to 1. Excel contains a built-in function (under the heading of “More Functions” – “information”) to determine whether a number is odd. The ISODD function returns a value of “True” if the reference cell is an odd number and false otherwise. Therefore, the following IF function can be used to calculate the parity bit value assuming we want even parity:

=IF(ISODD(J2),1,0)

The ISODD function tests whether the value in cell J2 is odd. If it is, the IF function evaluates to true and displays a 1 in column K. If the ISODD function is false, the IF function returns the value 0.

COLUMN L: The objective here is to calculate the parity bit value for odd parity. Odd parity means that there should be an odd number of bits, including the parity bit, that have a value of 1. Therefore, if the value in column J is even (there are an even number of bits in the 7-digit binary number representation of the plaintext character in that row) then the parity bit must be set to 1 in order to yield an odd number of bits with the value 1. For example, in row 2, the binary representation of the lowercase letter “a” is 1100001 which, as shown in Column J, contains an odd number of bits with a value of 1. Therefore, the parity bit for the lowercase letter “a” must be set to 0. Excel contains a built-in function (under the heading of “More Functions” – “information”) to determine whether a number is even. The ISEVEN function returns a value of “True” if the reference cell is an even number and false otherwise. Therefore, the following IF function can be used to calculate the parity bit value assuming we want odd parity:

=IF(ISEVEN(J2),1,0)

The ISEVEN function tests whether the value in cell J2 is even. If it is, the IF function evaluates to true and displays a 1 in column L so that the resulting 8-digit binary number contains an odd number of bits set to value of 1. If the ISEVEN function is false, the IF function returns the value 0 for the parity bit.

**Adjustment for special characters**:

Note that the five special characters (? ! % & ;) have only 6-digits to begin with (column B). Therefore, columns H and I duplicate each other. Consequently, the formula in column J must be adjusted to only sum the values for columns C through H. The formulas for the parity bit can then remain the same as used in the rows for the upper and lower case letters. If all characters are going to be represented by a string of 8 bits, then a leading 0 would be appended to the left of the code for each special character (i.e., the code for the ? would be 0111111 plus a parity bit.)

**10.6 The ABC Company is considering the following options for its backup plan:**

**1. Daily full backups:**

* **Time to perform backup = 60 minutes**
* **Size of backup = 50 GB**
* **Time to restore from backup = 30 minutes**

**2. Weekly full backups plus daily incremental backup:**

* **Same time, storage, and restoration as above to do a weekly backup on Friday, plus**
  + **Time to perform daily backup = 10 minutes**
  + **Size of daily backup = 10 GB**
  + **Time to restore each daily backup file = 5 minutes**

**3. Weekly full backups plus daily differential backup:**

* **Same time, storage, and restoration as above to do a weekly backup on Friday, plus**
  + **Time to perform daily backup = 10 minutes first day, growing by 5 minutes each day thereafter**
  + **Size of daily backup = 10 GB first day, growing by 10 GB each day**
  + **Time to restore differential backup file = 5 minutes first day, increasing by 2 minutes each subsequent day**

**Which approach would you recommend? Why?**

Solution: Management must weigh the trade-offs shown below.

Full daily backups take the most time to perform and require most storage, but in the event of a disaster have the quickest restore time.

Daily incremental backups on average take less time and use less storage than daily differential backups, but restoration is faster for the differential backups.

|  |  |  |  |
| --- | --- | --- | --- |
| **Type of Backup Plan** | **Time spent weekly to backup** | **Storage requirements** | **Time to Restore** |
|  |  |  |  |
| **Option 1:** Full Daily Backup | | | |
|  | 300 Minutes (5 days \* 60 minutes) | 250 GB (5 days \* 50 GB/day) | 30 Minutes to restore most recent full backup |
| **Total** | **300 Minutes** | **250 GB** | **30 Minutes** |
|  |  |  |  |
| **Option 2**: weekly full backup plus daily incremental backup | | | |
| Full Weekly Backup on Friday | 60 Minutes | 50 GB | 30 Minutes to restore last full backup |
| Daily Incremental Backup | 40 Minutes (4 days \* 10 minutes/day) | 40 GB (4 days \* 10 GB/day) | 5- 20 Minutes (5 minutes per day since last full backup) |
| **Total** | **100 Minutes** | **90 GB** | **35-50 Minutes** |
|  |  |  |  |
| **Option 3:** weekly full backup plus daily differential backup | | | |
| Full Weekly Backup | 60 Minutes | 50 GB | 30 Minutes to restore last full backup |
| Daily Differential Backup | 70 Minutes (10 minutes first day, increasing by 5 minutes/day – but on Friday, just make the full weekly backup) =  10+15+20+25=70 | 100 GB (10 GB for first day, 20 GB for second day, etc. but on Friday, just make the full weekly backup)=  10+20+30+40=100 | 5- 11 Minutes (5 minutes first day, 2 minutes more each subsequent day but on Friday, just make the full weekly backup) |
| **Total** | **130 Minutes** | **150 GB** | **35-41 Minutes** |

**10.7 Which control(s) would best mitigate the following threats?**

**a. The hours worked field in a payroll transaction record contained the value 400 instead of 40. As a result, the employee received a paycheck for $6,257.24 instead of $654.32.**

A limit check on hours worked. The limit would have to be higher than 40 (such as 55 – or whatever the company deemed appropriate) to allow for overtime, but would certainly catch the extra 0 added to the 40 hours worked.

**b. The accounts receivable file was destroyed because it was accidentally used to update accounts payable.**

All files should have header labels to identify their contents, and all programs should check these labels before processing transactions against the file.

There should also be a clearly marked external label to reduce the risk of an operator loading the wrong file.

**c. During processing of customer payments, the digit 0 in a payment of $204 was mistakenly typed as the letter “O.” As a result, the transaction was not processed correctly and the customer erroneously received a letter that the account was delinquent.**

A field check should be performed to check whether all characters entered in this field are numeric.

There should be a prompt correction and re-processing of erroneous transactions.

**d. </inst>A salesperson mistakenly entered an online order for 50 laser printers instead of 50 laser printer toner cartridges.**

A reasonableness test of quantity ordered relative to the product if 50 is an unusually large number of monitors to be ordered at one time.

Closed-loop verification to make sure that the stock number matches the item that is ordered.

**e. A 20-minute power brownout caused a mission-critical database server to crash, shutting down operations temporarily.**

An uninterruptible power system should be used to provide a reserve power supply in the event of power failure. The UPS should at a minimum allow enough time for the system to operated for a defined length of time and then, if necessary, power down in the event of an extended power outage.

Longer power outages are best handled by backup generators and real-time mirroring systems

**f. A fire destroyed the data center, including all backup copies of the accounts receivable files.**

FILES: A backup copy of the files should be stored off-site.

HARDWARE: A hot or cold site arrangement

BOTH: Real-time mirroring, so that when one site is down the other site(s) can pick up the slack.

A disaster recovery plan

Liability and business interruption insurance

1. **After processing sales transactions, the inventory report showed a negative quantity on hand for several items.**

A sign test of quantity on hand.

1. **A customer order for an important part did not include the customer’s address. Consequently, the order was not shipped on time and the customer called to complain.**

A completeness check to determine whether all required fields were filled in.

1. **When entering a large credit sale, the clerk typed in the customer’s account number as 45982 instead of 45892. That account number did not exist. The mistake was not caught until later in the week when the weekly billing process was run. Consequently, the customer was not billed for another week, delaying receipt of payment.**

Check digit verification on each customer account number

Or a validity check for actual customers.

1. **A visitor to the company’s Web site entered 400 characters into the five-digit Zip code field, causing the server to crash.**

A size check would prevent 400 characters from being entered into a field that allows for only 5 characters.

1. **Two traveling sales representatives accessed the parts database at the same time. Salesperson A noted that there were still 55 units of part 723 available and entered an order for 45 of them. While salesperson A was keying in the order, salesperson B, in another state, also noted the availability of 55 units for part 723 and entered an order for 33 of them. Both sales reps promised their customer next-day delivery. Salesperson A’s customer, however, learned the next day that the part would have to be back-ordered. The customer canceled the sale and vowed to never again do business with the company.**

Concurrent update controls protect records from errors when more than one salesman tries to update the inventory database by locking one of the users out of the database until the first salesman’s update has been completed.

1. **The warranty department manager was upset because special discount coupons were mailed to every customer who had purchased the product within the past 3 years, instead of to only those customers who had purchased the product within the past 3 months.**

A limit check based on the original sales date.

1. **The clerk entering details about a large credit sale mistakenly typed in a nonexistent account number. Consequently, the company never received payment for the items.**

Check digit verification on each customer account number

Or a validity check for actual customers

Or closed loop verification that returns the customer name associated with a customer number.

1. **A customer filled in the wrong account number on the portion of the invoice being returned with payment. Consequently, the payment was credited to another customer’s account.**

Turnaround documents should include account numbers on them.

1. **A batch of 73 time sheets was sent to the payroll department for weekly processing. Somehow, one of the time sheets did not get processed. The mistake was not caught until payday, when one employee complained about not receiving a paycheck.**

Batch totals would have caught this.

A record count would have indicated that one record was not processed.

Or a hash total (sum of the employee numbers).

**q. Sunspot activity resulted in the loss of some data being sent to the regional office. The problem was not discovered until several days later when managers attempted to query the database for that information.**

Parity checks and checksums will test for data transmission errors.

**10.8 MonsterMed Inc. (MMI) is an online pharmaceutical firm. MMI has a small systems staff that designs and writes MMI’s customized software. The data center is installed in the basement of its two-story headquarters building. The data center is equipped with halon-gas fire suppression equipment and an uninterruptible power supply system.**

**The computer operations staff works a two-shift schedule, five days per week. MMI’s programming staff, located in the same building, has access to the data center and can test new programs and program changes when the operations staff is not available. Programmers make changes in response to oral requests by employees using the system. Since the programming staff is small and the work demands have increased, systems and programming documentation is developed only when time is available. Backups are made whenever time permits. The backup files are stored in a locked cabinet in the data center. Unfortunately, due to several days of heavy rains, MMI’s building recently experienced serious flooding that destroyed not only the computer hardware but also all the data and program files that were on-site.**

**a. Identify at least five weaknesses in MonsterMed Inc.’s backup and DRP procedures.**

1. No written backup.
2. No written disaster recovery plan.
3. Backups are not done on a regular basis.
4. Restoration of backups is not tested.
5. Systems documentation is prepared when someone has the time to do it; consequently, documentation will be incomplete and not current.
6. The programming staff has access to the computer room without supervision of the operations staff. The programmers could alter the data files or operational programs.
7. The location of the computing facility in the basement increases the risk of damage due to flooding.

8. No written request, approval process, or testing process for systems changes

**b. Evaluate change controls at MonsterMed Inc.**

1. There does not appear to be a separate testing and development system, so changes are made directly in the production system.

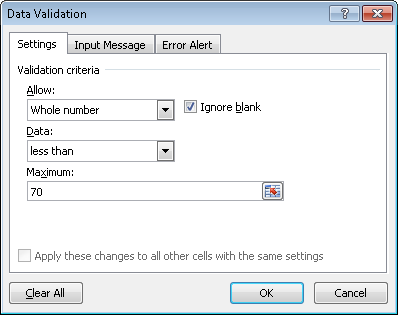
2. Change requests are made orally, with no formal approval or documentation.

(Adapted from CMA Exam)

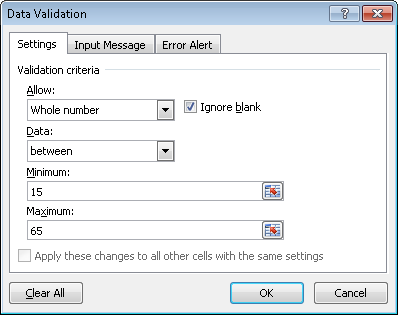
**10.9** **Excel Problem**

**Create data validation rules in a spreadsheet to perform each of the following controls:**

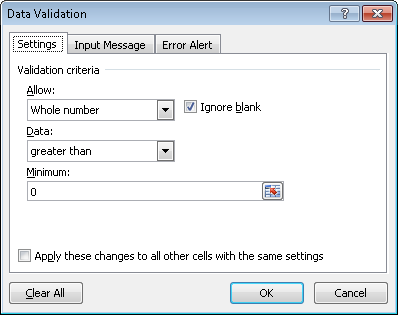
1. **Limit check – that values in the cell are < 70**



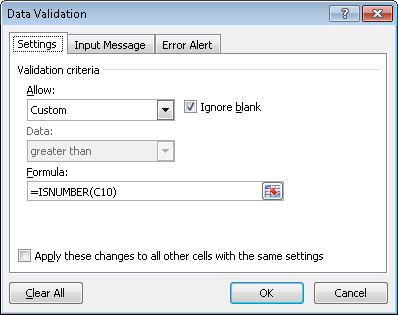
1. **Range check – that values in the cell are between 15 and 65**



1. **Sign check – that values in the cell are positive**

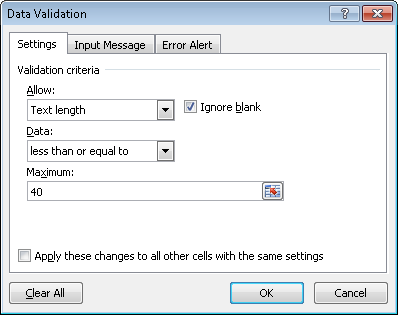


1. **Field check – that values in a cell are only numeric**

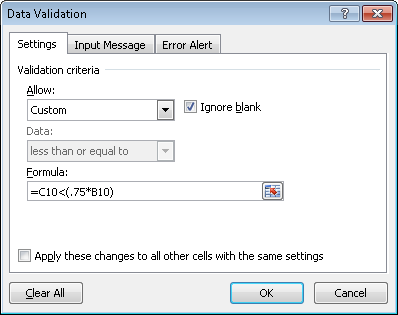


The ISNUMBER function tests whether the cell contains only numeric data.

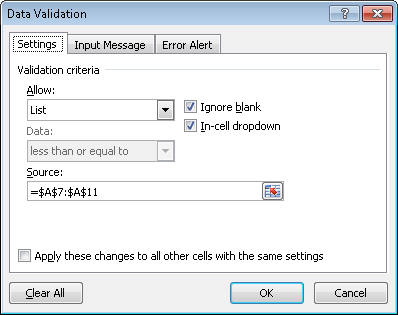
1. **Size check – that cell accepts no more than 40 characters of text**



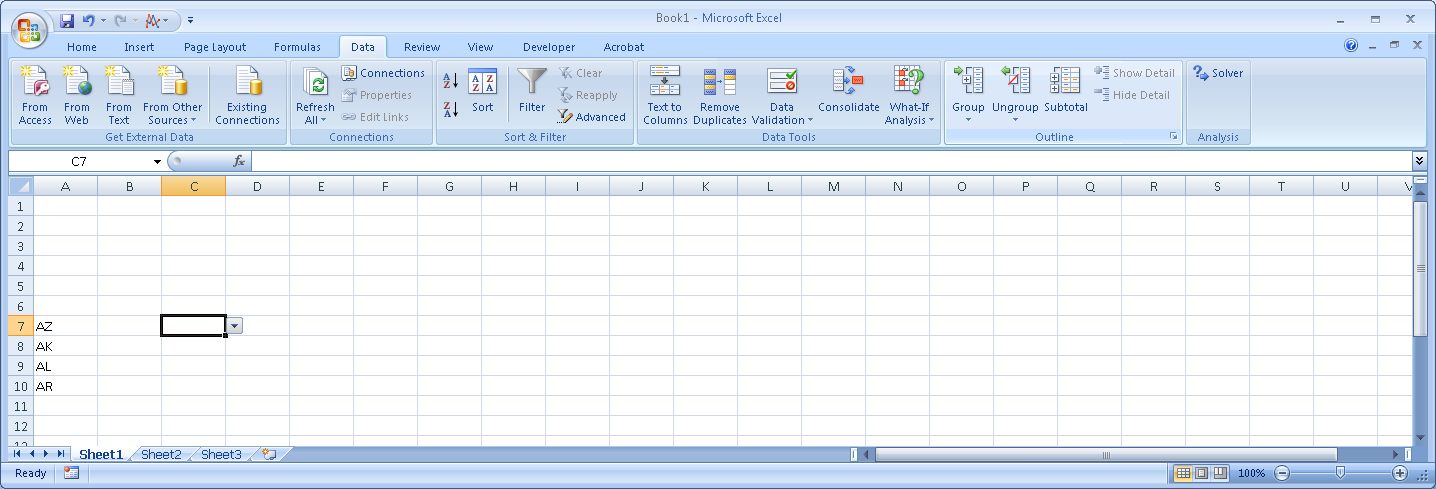
1. **Reasonableness check – that cell’s value is less than 75% of cell to its left**



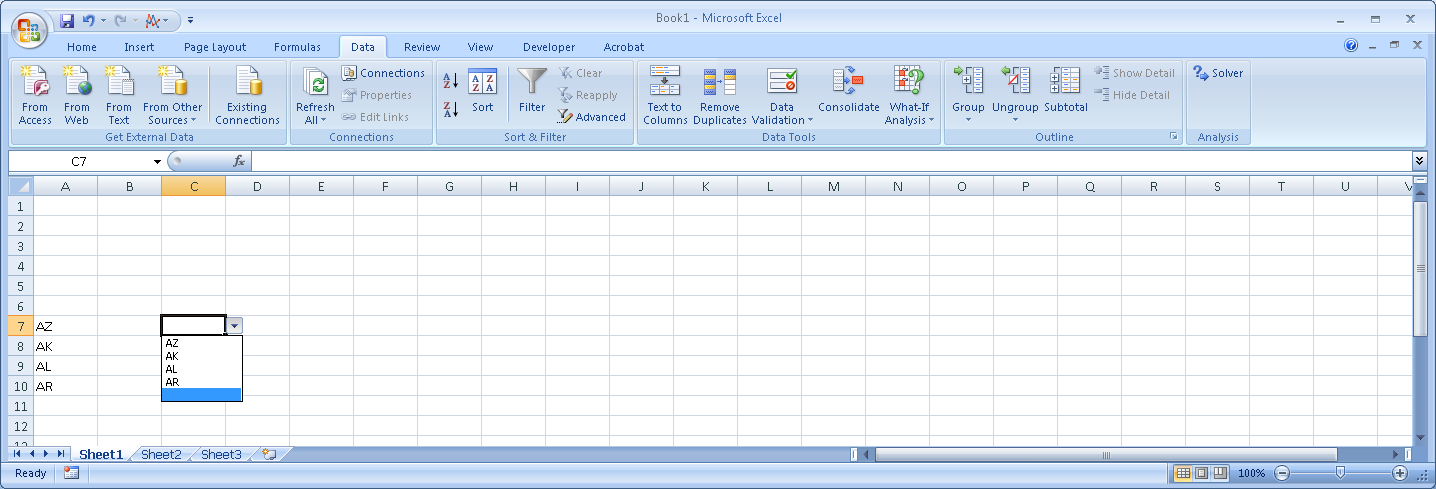
1. **Validity check – that a value exists in a list of allowable values**



If the preceding data validation rule was applied to cell C7, the spreadsheet would look like this:



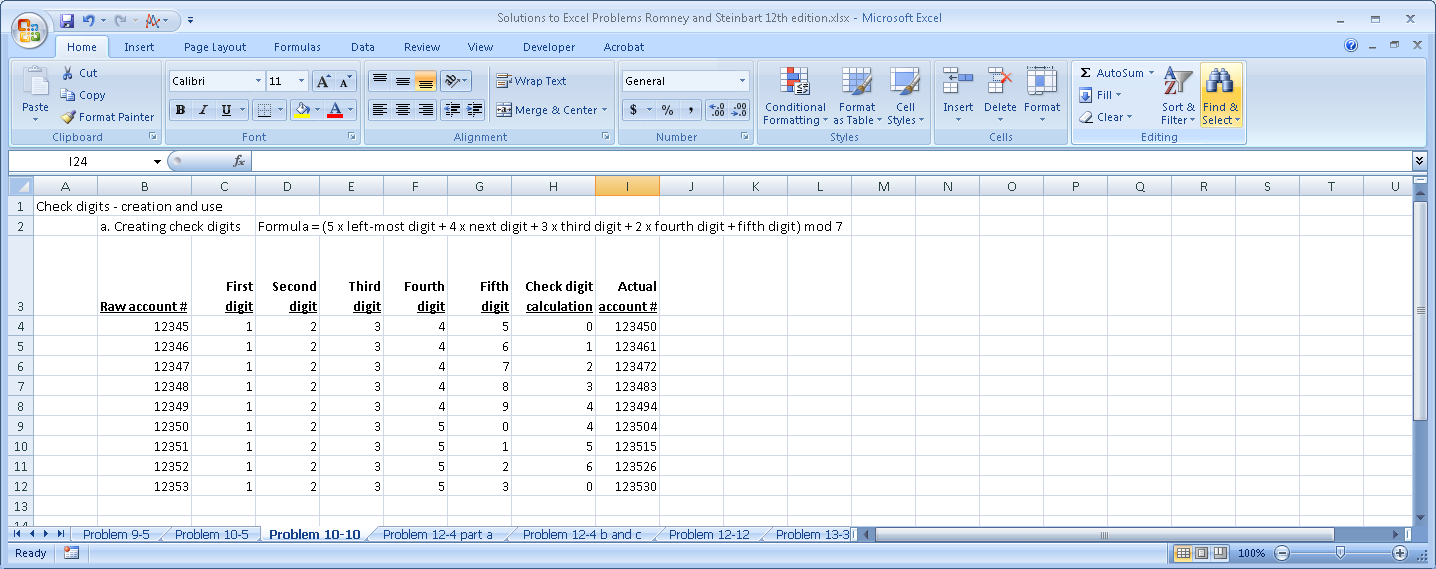
And clicking the drop-down arrow would display the following:



**10.10 Excel Problem**

**Creating and testing check digits.**

**a. Create a spreadsheet that will take as input a five-digit account number and calculate a check digit using this formula: (5 x left-most digit + 4 x next digit + 3 x third digit + 2 x fourth digit + fifth digit) modulus division by 7. (Modulus division returns the remainder – for example: 11 modulus division by 3 = 2). The check digit then becomes the 6th (right-most) digit in the account number. Your spreadsheet should look like this:**



Explanation: the formula for the check digit calculation is =MOD(((5\*C4)+(4\*D4)+(3\*E4)+(2\*F4)+G4),7). The MOD function is one of Excel’s built-in “Math&Trig” functions. It takes two arguments: the number you are dividing, and the divisor. In this case, the number is a formula (5 x the first digit in cell C4 plus 4 x the second digit in cell D4 plus 3 x the third digit in cell E4 plus 2 x the second digit in cell F4 plus the last digit from cell G4). This result is then divided by 7, and the MOD function returns the remainder. Thus for the first row, the formula yields: (5 x 1) + (4 x 2) + (3 x 3) + (2 x 4) + 5 = 35. Dividing 35 by 7 yields 5 with a remainder of 0. Therefore, the MOD function returns a value of 0 for the check digit. Appending the zero to raw account number yields the actual account number of 123450.

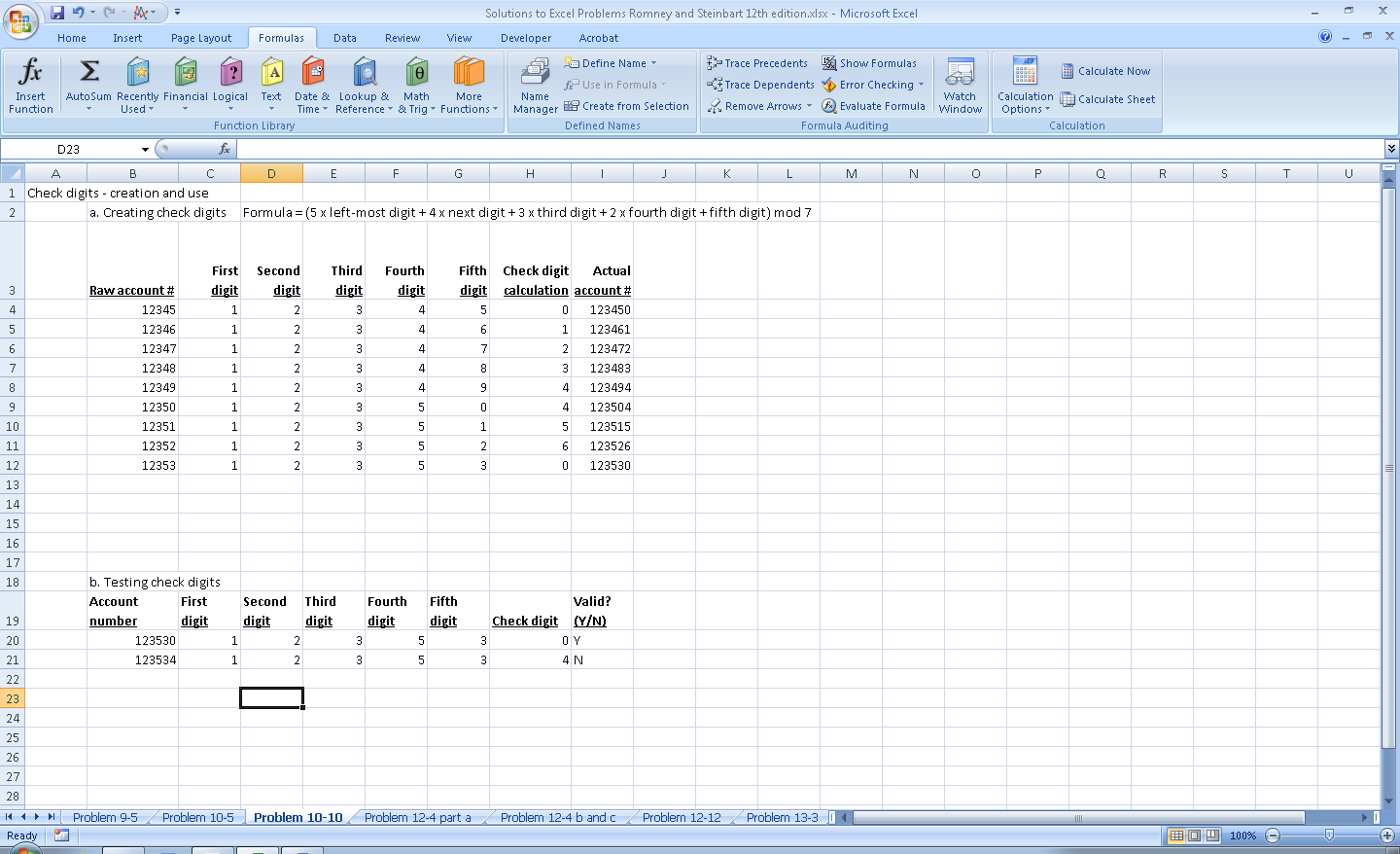
Students should use the text formulas (LEFT and RIGHT) plus the VALUE formula to parse the raw account number from column B in order to automatically fill in columns C through G as follows:

* Column C: =VALUE(LEFT(B4)). The LEFT function with one argument is used to return the left-most digit from reference cell (B4). The result is a text value of 1. Then the VALUE function converts that text into the number 1.
* Column D: =VALUE(LEFT(RIGHT(B4,4))). The combination of LEFT and RIGHT functions is used to return the second digit from the left in the reference cell (B4). The RIGHT function can take two arguments: the cell containing the numeric value to be manipulated (in this case B4) and the number of digits, beginning with the rightmost one, to return. In this case, it returns the 4 right-most digits: 2345. Next, the LEFT function lops off the left-most digit in that string, yielding text string of “2”. Finally, the VALUE function converts that text into the number 2.
* Column E: =VALUE(LEFT(RIGHT(B4,3))). The combination of LEFT and RIGHT functions is used to return the third digit in reference cell (B4). The RIGHT function can take two arguments: the cell containing the numeric value to be manipulated (in this case B4) and the number of digits, beginning with the rightmost one, to return. In this case, it returns the 3 right-most digits: 345. Next, the LEFT function lops off the left-most digit in that string, yielding text string of “3”. Finally, the VALUE function converts that text into the number 3.
* Column F: =VALUE(LEFT(RIGHT(B4,2))). The combination of LEFT and RIGHT functions is used to return the fourth digit in the reference cell (B4). The RIGHT function can take two arguments: the cell containing the numeric value to be manipulated (in this case B4) and the number of digits, beginning with the rightmost one, to return. In this case, it returns the 2 right-most digits: 45. Next, the LEFT function lops off the left-most digit in that string, yielding text string of “4”. Finally, the VALUE function converts that text into the number 4.
* Column G: =VALUE(RIGHT(B4)). The RIGHT function with one argument returns the rightmost character from the reference cell (B4). In this case, it returns the text string of “5”. Then the VALUE function converts that text into the number 5.

Finally, the actual account number in column I can be created using the CONCATENATE and VALUE functions:

* The CONCATENATE function is one of Excel’s built-in text functions that appends two strings together. Thus, in cell I4, the function CONCATENATE(B4, H4) would append the value in cell H4 (which is the calculated check-digit of 0) to the value in cell B4 (the raw account number 12345) yielding the string 123450.
* The VALUE function then transforms that text string of 123450 into the number 123450.

**b. Add another panel to the spreadsheet that takes as input a six-digit account number and uses the check digit formula in part a to test whether or not the account number is valid. Your solution should look like this:**



Solution: This formula in the “Valid? (Y/N)” column will test any six-digit account number:

=IF(H20=MOD(((C20\*5)+(D20\*4)+(E20\*3)+(F20\*2)+G20),7),"Y","N")

Ideally, all a user should need to do is input a six-digit account number in the “Account number” column and then the spreadsheet will display the individual digits in the appropriate columns. This requires the following formulas to parse the six-digit account number entered in cell B20 (you can copy these formulas down for as many rows as desired):

“First digit” column: =VALUE(LEFT(B20))

“Second digit” column: =VALUE(LEFT(RIGHT(B20,5)))

“Third digit” column: =VALUE(LEFT(RIGHT(B20,4)))

“Fourth digit” column: =VALUE(LEFT(RIGHT(B20,3)))

“Fifth digit” column: =VALUE(LEFT(RIGHT(B20,2)))

“Check digit” column: =VALUE(RIGHT(B20))

Alternatively, if you only want a two-column display with the account number and the “Valid? (Y/N)” column, you could modify the formula in the “Valid? (Y/N)” column so that the cell references were replaced as follows:

=IF(H20=MOD(((VALUE(LEFT(B20))\*5)+( VALUE(LEFT(RIGHT(B20,5)))\*4)+( VALUE(LEFT(RIGHT(B20,4)))\*3)+( VALUE(LEFT(RIGHT(B20,3)))\*2)+ VALUE(LEFT(RIGHT(B20,2)))),7),"Y","N")

**10. 11 For each of the following scenarios, determine whether the company’s current backup procedures enable it to meet its recovery objectives and explain why:**

**a. Scenario 1:**

* **Recovery point objective = 24 hours**
* **Daily backups at 3:00 am, process takes 2 hours**
* **Copy of backup tapes picked up daily at 8:00 am for storage off-site**

Solution: No. Many companies make two backup copies – one to keep locally and one to store offsite. If a fire or similar event destroyed the data center on a weekday before 8:00 a.m., both copies of the most recent daily backup tapes would be destroyed because the disaster happened before the second copy was picked up for offsite storage. For example, assume that a fire happened Wednesday morning at 7:00 a.m. Both copies of Tuesday night’s back-up tape would have been destroyed. It does have a copy of Monday night’s backup stored off-site. But this means it would have lost all data since the backup that was made at 3:00 am on Tuesday morning. Consequently, the company would be missing 28 hours of data (all transactions that happened between 3:00 am Tuesday and 7:00 am on Wednesday), which is more than its recovery point objective of 24 hours.

**b. Scenario 2: Company makes daily incremental backups Monday-Saturday at 7:00 pm each night. Company makes full backup weekly, on Sunday at 1:00 pm.**

* **Recovery time objective = 2 hours**
* **Time to do full backup = 3 hours**
* **Time to restore from full backup = 1 hour**
* **Time to make incremental daily backup = 1 hour**
* **Time to restore each incremental daily backup = 30 minutes**

Solution: No. If a disaster happened any time after 7:00 pm on Wednesday, it would take more than 2 hours to completely restore all backups:

Time to restore from Sunday’s full backup = 1 hour

Time to restore Monday’s incremental backup = 30 minutes

Time to restore Tuesday’s incremental backup = 30 minutes

Time to restore Wednesday’s incremental backup = 30 minutes

Total time to restore = 2.5 hours

**c. Scenario 3: Company makes daily differential backups Monday-Friday at 8:00 p.m each night. Company makes full backup weekly, on Saturdays, at 8:00 am.**

* **Recovery time objective = 6 hours**
* **Time to do full backup = 4 hours**
* **Time to restore from full backup = 3 hours**
* **Time to do differential daily backups = 1 hour on Monday, increasing by 30 minutes each successive day**
* **Time to restore differential daily backup = 30 minutes for Monday, increasing by 15 minutes each successive day**

Solution: Yes. Even if a disaster happened early Saturday morning (say at 3:00 am) the company would not have yet done a full backup, but would have completed its final differential backup Friday night. Therefore, full restoration would take:

Time to restore from last Saturday’s full backup = 3 hours

Time to restore Friday’s differential backup = 1 hour 30 minutes

Total time to restore = 4.5 hours

The total time of 4.5 hours is less than the RTO of 6 hours.

If a disaster happened earlier in the week, the company would take even less time to restore. For example, if a fire destroyed the data center Wednesday morning, the company would have to restore the previous Saturday’s full backup plus Tuesday night’s differential backup:

Time to restore from last Saturday’s full backup = 3 hours

Time to restore Friday’s differential backup = 45 minutes

Total time to restore = 3.75 hours

which is less than the RTO of 6 hours.

**SUGGESTED ANSWERS TO THE CASES**

**Case 10-1 Ensuring Systems Availability**

**The Journal of Accountancy (available at** [**www.aicpa.org**](http://www.aicpa.org)**) has published a series of articles that address different aspects of disaster recovery and business continuity planning:**

1. Gerber, J. A., and Feldman, E. R. 2002. “Is Your Business Prepared for the Worst?” *Journal of Accountancy* (April): 61-64.
2. McCarthy, E. 2004. “The Best-Laid Plans,” *Journal of Accountancy* (May): 46-54.
3. Myers, R. 2006. “Katrina’s Harsh Lessons,” *Journal of Accountancy* (June): 54-63.
4. Phelan, S., and Hayes, M. 2003. “Before the Deluge – and After,” *Journal of Accountancy* (April): 57-66.

**Read one or more of the following articles that your professor assigns plus section DS4 of COBIT version 4.1 (available at** [**www.isaca.org**](http://www.isaca.org)**) to answer the following questions:**

1. **What does COBIT suggest as possible metrics for evaluating how well an organization is achieving the objective of DS4? Why do you think that metric is useful?**

|  |  |  |
| --- | --- | --- |
| **Proposed Metric** | | **Why useful** |
| Number of hours lost per user per month due to unplanned outages | * High level measure of availability reflecting overall success * Need to subtract any planned downtime for upgrades to get accurate metric |
| Percent of availability SLAs met | * If referring to vendors, this measures how well they meet obligations * If referring to company, measures how well it is fulfilling its contractual obligations |
| Number of business-critical processes relying on IT not covered by IT continuity plan | * Focus on critical business processes for which there is no DRP or BCP. This is a warning sign of potential risks. |
| Percent of tests that achieve recovery objectives | * Evaluates performance of testing the DRP and BCP (detective measure that identifies areas in need of improvement) |
| Frequency of service interruption of critical systems | * Another measure of overall performance. Helps interpret the hours lost metric – (e.g., did the organization have just one or two major problems or many smaller ones?) |
| Elapsed time between tests of any given element of IT continuity plan | * Indicates areas in need of testing |
| Number of IT continuity training hours per year per relevant employee | * Measure of preparedness |
| Percent of critical infrastructure components with automated availability monitoring | * Measure of extent of usage of cost-effective proactive availability controls |
| Frequency of review of IT continuity plan | * Measure of preparedness and how well the DRP and BCP are maintained |

1. **For each article assigned by your professor, complete the following table, summarizing what each article said about a specific COBIT control objective (an article may not address all 10 control objectives in DS4):**

Solution: Answers will vary, but should include at least the following:

Gerber, J. A., and Feldman, E. R. 2002. “Is Your Business Prepared for the Worst?”

|  |  |
| --- | --- |
| **COBIT Control Objective** | **Points discussed in article** |
| DS4.1 | Lists who should be involved in developing the framework and plan  Don’t overlook key external parties and contact methods |
| DS4.2 | Who should be involved in developing the framework and plan |
| DS4.3 | Discusses how details of the plans will differ depending upon the nature of the organization’s business operations |
| DS4.4 |  |
| DS4.5 | Need to do simulations and other tests |
| DS4.6 | Practice the plans and everyone’s roles |
| DS4.7 | Make sure everyone understands the plan |
| DS4.8 | Plans should specify how to recover from the disaster and resume operations |
| DS4.9 |  |
| DS4.10 |  |

McCarthy, E. 2004. “The Best-Laid Plans,” *Journal of Accountancy* (May):

|  |  |
| --- | --- |
| **COBIT Control Objective** | **Points discussed in article** |
| DS4.1 |  |
| DS4.2 |  |
| DS4.3 | How to prioritize what needs to be protected and how to protect |
| DS4.4 | Need to update the plan |
| DS4.5 | How to test plans – specific things to do/consider for scenario tests |
| DS4.6 | Review the test results with employees to identify what worked, what didn’t |
| DS4.7 |  |
| DS4.8 |  |
| DS4.9 | Checklist of how to do backups, where to store, etc. |
| DS4.10 | Importance of periodically reviewing the plans and updating |

Myers, R. 2006. “Katrina’s Harsh Lessons,” *Journal of Accountancy* (June):

|  |  |
| --- | --- |
| **COBIT Control Objective** | **Points discussed in article** |
| DS4.1 | Reviews different types of plans and what each contains |
| DS4.2 |  |
| DS4.3 |  |
| DS4.4 |  |
| DS4.5 | Need to test the plan at least annually |
| DS4.6 | Divide responsibilities across employees and practice |
| DS4.7 | Importance of communications procedures – and specific recommendations of how to ensure you can do this |
| DS4.8 | Specific steps for how to recover data after floods, fires, etc. |
| DS4.9 | Examples of why you need off-site backup copies |
| DS4.10 |  |

Phelan, S., and Hayes, M. 2003. “Before the Deluge – and After,”

|  |  |
| --- | --- |
| **COBIT Control Objective** | **Points discussed in article** |
| DS4.1 | Involve senior management in developing the plans |
| DS4.2 | Discusses hot sites and other issues about planning to replace the infrastructure  Examples of the benefits of having a plan so can be prepared |
| DS4.3 | Specific examples of the kinds of information assets that need to backup |
| DS4.4 |  |
| DS4.5 |  |
| DS4.6 | Communication methods discussed |
| DS4.7 |  |
| DS4.8 | Detailed side-bar on how to actually recover data/information in various situations |
| DS4.9 |  |
| DS4.10 |  |

**Case 10-2 Change Controls**

**Read section AI6 in version 4.1 of COBIT (available at** [**www.isaca.org**](http://www.isaca.org)**) and answer the following questions:**

1. **What is the purpose of each detailed control objective – why is it important?**

**AI6.1 Change Standards and Procedures**

Set up formal change management procedures to handle in a standardised manner all requests (including maintenance and patches) for changes to applications, procedures, processes, system and service parameters, and the underlying platforms.

Reason it is important

* Unauthorized changes can introduce malware and weaken segregation of duties.
* Failure to formally document changes makes it difficult to recover functionality after a disaster.

**AI6.2 Impact Assessment, Prioritisation and Authorisation**

Assess all requests for change in a structured way to determine the impact on the operational system and its functionality. Ensure that changes are categorised, prioritised and authorised.

Reason it is important

* Proactive analysis of proposed changes reduces the risk of making changes that negatively affect system performance and availability.

**AI6.3 Emergency Changes**

Establish a process for defining, raising, testing, documenting, assessing and authorising emergency changes that do not follow the established change process.

Reason it is important

* Emergency changes occur in response to problems or incidents. It is often important to resolve the problem quickly by implementing a change without going through the formal change control management process.
* Once the problem has been solved or the crisis is over, it is important to go back and test the changes for any other unanticipated side effects.
* It is also important to document the change, so that in the event of a subsequent incident the system can be properly restored.

**AI6.4 Change Status Tracking and Reporting**

Establish a tracking and reporting system to document rejected changes, communicate the status of approved and in-process changes, and complete changes. Make certain that approved changes are implemented as planned.

Reason it is important

* Employees will not abide by change control procedures if they do not receive prompt feedback on requests.

**AI6.5 Change Closure and Documentation**

Whenever changes are implemented, update the associated system and user documentation and procedures accordingly.

Reason it is important

* Changes need to be documented so that they can be replicated, if necessary, in the event of future problems.

1. **How is each of the suggested metrics useful?**

|  |  |  |
| --- | --- | --- |
| **Suggested metric** | | **Why useful** |
| Number of disruptions or data errors caused by inaccurate specifications or incomplete impact assessments | * Overall measure of effectiveness of change controls in preventing problems |
| Amount of application rework caused by inadequate change specification | * Another outcome measure of overall effectiveness of the change control process |
| Reduced time and effort required to make changes | * Positive outcome measure reflecting the overall goal of change control |
| Percent of total changes that are emergency fixes | * Measures compliance with change control process. A high number of emergency changes is evidence that people may be “gaming” the system, claiming something is an emergency in order to avoid formal change control. Helpful in measuring compliance with DS6.3 |
| Percent of unsuccessful changes to the infrastructure due to inadequate change specifications | * Negative outcome measure of compliance with DS6.2 |
| Number of changes not formally tracked, reported, or authorized | * Negative outcome measure of overall effectiveness of change control process, measures compliance with DS6.1 |
| Number of backlogged change request | * Efficiency measure for DS6.4 |
| Percent of changes recorded and tracked with automated tools | * Compliance with change control processes requires timely feedback on requests. This metric assesses efficiency of DS6.4 |
| Percent of changes that follow formal change control process | * Overall measure of effectiveness of change control; also useful to assess DS6.3 |
| Ratio of accepted to refused change requests | * Feedback to employees; relevant to DS6.4 |
| Number of different versions of each business application or infrastructure being maintained | * Measures compliance with change control process – higher scores here suggest lack of standard procedures and numerous ad hoc changes |
| Number and type of emergency changes to the infrastructure components | * Measure of overall compliance with formal change control process; also relevant to DS6.3 |
| Number and type of patches to the infrastructure components | * Patches are planned changes, so this measures preventive actions taken |