LESSON 6 - BASIC PROGRAMMING - DO LOOP, KEY INPUT, SELECT CASE

DO Loops: Type the following programs in. Run them and learn how they work. You will use these to do the original programs you have to create.

Ex. 1)
```
! Infinite Loop
Do
        Print "Happy Birthday";
        Print " And many happy returns.";
Loop
End
```
(The DO loop will execute the block of statements between Do and Loop forever. There are 3 ways to end a DO loop. Attach a condition to DO, a condition to Loop, or have and Exit Do statement.

Ex. 2)
```
! First condition - attaching a condition to Do
Let x = 2
Do while x < 1000
        Print x;
        let x = x + 2
Loop
End
```

Ex. 3)
```
!Second condition - attaching a condition to Loop
Let x = 2
Do
        Print x;
        let  x = x + 5
Loop while x < 1000
End
```

Ex. 4)
```
! Checking to see if we need to continue?
Input prompt " What is your age?": age
Input prompt " What year were you born in?": year
Do
        Print "The year is ";year;" and you are"; age; "years old."
        let age = age + 1
        let year = year + age
        Print "Shall I continue?"
        Input answer$
Loop while answer$ = "yes"
Print "It was nice doing business with you!"
End
```

Ex. 5)
```
! Using an Exit Do
Input prompt " What is your age?": age
Input prompt " What year is it now?": year
Do
        Print "The year is ";year;" and you are"; age; "years old."
        let age = age + 1
        let year = year + 1
        Print "Shall I continue?"
        Input answer$
        If answer$ = "no" then Exit Do
Loop
Print "It was nice doing business with you!"
End
```

GET KEY Command:

Sometimes we wish to have the user determine what the program does next, indicating his/her choice by pressing a single key. The statement GET KEY z waits until the user presses a key, then translates that key into its corresponding number in computer language and assigns that number to the variable z . If the character chosen is an ASCII character, the ASCII number is assigned. Try the following using the GET KEY command.

Ex. 6)
```
! User presses a key to stop program
Do
        Get Key k
        Print k
Loop until k = 32                    ! The # 32 is the space bar
End
```

SELECT CASE

If all the choices in a decision structure are based on the value of a single expression, a SELECT CASE statement is convenient. The expression to be evaluated is in the SELECT CASE statement. After it is evaluated, the appropriate CASE is chosen. Try the following.

Ex. 7)
```
! Identifying the number types from 0 to 10
For  x = 0 to 10
Select case  x
Case 2
        Print x
        Print "Even prime"
Case 3,5,7
        Print x
        Print "Odd prime"
Case 1,4,9
        Print x
        Print "Perfect square"
Case else
        Print x
        Print "composite, not a square"
End Select
Next x
End
```

(By using "to" you can specify a range of values. By using is you can make a comparison.)

Ex. 8)
```
! Do you want to be a millionaire?
Do
        Input prompt "What is your salary?": s
        Select case s
        Case 0 to 999999
                Print "Not a millionaire yet."
        Case  is > 999999
                Print " A millionaire!!!!"
        Case else
                Print " I think you made an error or you need to get a job."
        End Select
        Print "Shall I continue?"
        Input answer$
        If answer$ = "no" then Exit Do
Loop
Print "Be sure to give some of that money to the poor!!"
End
```

Assignment:

Create a word processing document "Lesson 6" with a proper heading for math that will be where you transfer the following programs to be printed out.

1. Create a program that fills the screen with a message using a DO loop. (Hint - use program 1 as a gui Start your program with your name and room number on the first line and the name of the program on the second. Save your program as **Message. Have it checked by me before copying it to the word processing document. Put it into the word processing document Lesson 6.**

2. Create a program that will print the multiples of 4 less than 10000000000000. (Hint - use programs 2 or 3 as guides) Start your program with your name and room number on the first line and the name of the program on the second. Save your program as **Multiples of 4. Have it checked by me before copying it to the word processing document. Put it into the word processing document Lesson 6.**

3. Create a program that will ask a person his/her weight, how many pounds he/she would like to gain each year, and what year it is. Then have it show his/her weight each year until he/she wants to stop the program. (Hint - use program 5 as a guide.) Save your program as **Weight Year by Year.Have it checked by me before copying it to the word processing document. Put it into the word processing document Lesson 6.**

4. Create a program that will find the square root of any number that a user enters. Use a DO loop and us the GET KEY command to stop the program. Let the key that stops the program be the RETURN key. You will have to use program 6 to find the ASCII number for the RETURN key and to finish the program below.

```
! Finding the square root of any number
Do
        Input prompt "Enter a number you wish to find the square root of.": n
        Print  "The square root of  "; n "  is " ; sqr(n)
.
.
.
```

Save your program as **Square Root 2. Have it checked by me before copying it to the word processing document. Put it into the word processing document Lesson 6.**

5. Create a program using **SELECT CASE.** You can make it on any topic, but it must follow the structure of example 8. You can have more cases than example8. Save your program as **CASES. Have it checked by me before copying it to the word processing document. Put it into the word processing document Lesson 6.**