KEMENTERIAN PENDIDIKAN MALAYSIA

POLITEKNIK MALAYSIA
METrO Tasek Gelugor

POLITEKNIK METrO TASEK GELUGOR
NO 25, LORONG KOMERSIAL 2,
PUSAT KOMERSIAL TASEK GELUGOR
13300 TASEK GELUGOR
PULAU PINANG

DEPARTMENT OF INFORMATION
AND COMMUNICATION TECHNOLOGY

DFP30243:
OBJECT ORIENTED PROGRAMMING

NAME:	MUHAMMAD AFIQ MUHAIMIN BIN MOHD ZAINI
CLASS :	DDT3A
REGISTRATION NO :	32DDT20F2029
LECTURER :	HAZLEENA BINTI OSMAN
NAME OF TASK:	LAB ACTIVITY 1

LAB ACTIVITY 1:
Introduction to Java Programming

Learning Outcomes:

By the end of this laboratory session, you should be able to:

1. Identify the features of Java programming language.
2. Apply programming style and documentation in Java.
3. Write, compile and run the Java programs.
4. Troubleshoot the source code to identify the errors in Java.

Hardware/Software: Computer with JDK latest version.

Activity 1A

Activity Outcome : Identify the anatomy of the Java Program.

Procedure:

1. Analyze the HelloWorldApp program
2. Complete the following table with the correct line number and code of Java program:

```
1. /* Activity 1A */

2. class HelloWorldApp {
3.       public static void main(String[] args)
4.       {
5.             System.out.println("Hello World!");
6.       }
7. }
```

No	Components	Line No	Corresponding Code
1.	Comments	1.	`/* Activity 1A */`
2.	Class name	2.	`class HelloWorldApp`
3.	Access modifier	3.	`public`
4.	Argument	5.	`("Hello World!");`
5.	Static method	3.	`public static void main(String[] args)`

Activity 1B

Activity Outcome: Identify programming style and documentation in Java.Write java program with programming style and documentation.

Procedure:

Step 1: Open Notepad and type the following code:

```
/* Activity 1B */

class Act1B     //  define class name
{ // open curly brace the beginning of class block

      // main program where the program start
      public static void main(String[]
      args)
      {     // open curly brace for main block
            //statement to print string
            System.out.println("Hello, my name is  <YOUR
            NAME>.");System.out.println("\tI'm from_.");
      } //close curly brace for main block
} //end of class with close curly brace for body of block class
```

Step 2: Save, compile, and run the program. Save the program as `Act1B.java`. Write the outputin the area below.

Edited Code

```
/* Activity 1B */

class Act1B // define class name
{ // open curly brace the beginning of class block

// main program where the program start
public static void main(String[] args)
{   // open curly brace for main block
//statement to print string
    System.out.println("Hello, my name is Afiq."); System.out.println("\tI'm from Terengganu.");
} //close curly brace for main block
} //end of class with close curly brace for body of block class
```

Output:

```
nment\Lab Activity 1> java Act1B
Hello, my name is Afiq.
        Iâ??m from Terengganu.
```

Activity 1C

Activity Outcome: Identify programming style and documentation in Java program.Write java program with programming style and do documentation.

Procedures :

Step 1: Open Notepad and type the following code:

```
/* Activity 1C */
public class Act3F { public static void main(String[] args) {
int number1 = 11; int number2 = 22; int number3 = 33; int number4 =
44;int number5 = 55; int sum;
sum = number1 + number2 + number3 + number4 +
number5;System.out.print("The sum is ");
System.out.println(sum); }}
```

Step 2: Add the appropriate comments and comment style, proper indention and spacing and block styles

Edited Code

```
/* Activity 1C */
public class Act1C //Defining Class Name
{ //Open curly brace beginning of class block
//Main program where the program start
public static void main(String[] args)
{   //Open curly brace for main block
    int number1 = 11;   //Declare int variable number1 to 11
    int number2 = 22;   //Declare int variable number2 to 22
    int number3 = 33;   //Declare int variable number3 to 33
    int number4 = 44;   //Declare int variable number4 to 44
    int number5 = 55;   //Declare int variable number5 to 55
    int sum;            //Declare int sum
    //Formula for calculating sum of 5 numbers
    sum = number1 + number2 + number3 + number4 + number5;
    //Statement to print sum of 5 numbers
    System.out.print("The sum is "); System.out.println(sum);
}   //Close curly brace for main block
}   //Close curly brace beginning of class block
```

Step 3: Save, compile, and run the program. Save the program as `Act1C.java`. Write the outputin the area below.

Output:

```
nment\Lab Activity 1> java Act1C
The sum is 165
```

Activity 1D

Activity Outcome: Identify programming errors in Java program.Identify the errors and modify the program to fix the error(s).

```
/* Activity 1D */

Class Act1D
{
public static void Main(String[]
args)[
      System.out.println ("I Love Java Programming!”);
      System.out.println ("I know how to write Java
      Program!);System.out.print (“It is simple and
      easy.”)
)
```

Edited Code

```
/* Activity 1D */
class Act1D{
public static void main(String[] args)
{
      System.out.println ("I Love Java Programming!");
      System.out.println ("I know how to write Java Program!");
      System.out.println ("It is simple and easy.");
}
}
```

Output

```
nment\Lab Activity 1> java Act1D
I Love Java Programming!
I know how to write Java Program!
It is simple and easy.
```

Activity 1E (CLO2)

Activity Outcome: Identify programming errors in Java program.Identify the errors and modify the program to fix the error(s).

```
/* Activity 1E //

class Act1E
{
public static void Main(String[] args)
{
      double price1=50.65;
      double total =
      price1/0;
      System.out.println ("Total price is "+total);

}
```

Write the correct program below:

Program:

```
/* Activity 1E */
class Act1E
{
public static void main(String[] args){

double price1 = 50.65;
double total = price1;
System.out.println("Total price is " +total);
}
}
```

Output

```
nment\Lab Activity 1> java Act1E
Total price is 50.65
```

Activity 1F (CLO2)

Activity Outcome: Identify programming errors in Java.
Identify syntax errors in Java, implements programming style and documentation.

Procedures :

Step 1 : Open Notepad and type the following code:

```
/* Activity 1F */

Class Act3I {
      public Static void main(String[] args) {
            Sytem.out.prntln("Hello World!");
      }
}
```

Step 2 : Save the program as `Act1F.java` in the working directory, compile and run the program.

Step 3 : Observe the output.

```
nment\Lab Activity 1> javac .\Act1F.java
.\Act1F.java:3: error: class, interface, enum, or record expected
Class Act3I {
^
.\Act1F.java:4: error: class, interface, enum, or record expected
public Static void main(String[] args) { Sytem.out.prntln("Hello World!");
       ^
.\Act1F.java:5: error: class, interface, enum, or record expected
}
^
3 errors
```

Step 4: Identify the syntax error and correct the syntax errors.

Edited Code:

```
/* Activity 1F */

class Act3I {
public static void main(String[] args){
    System.out.println("Hello World!");
}
}
```

Step 5: Save your program, compile and run the program again.

Output:

```
nment\Lab Activity 1> java Act1F
Hello World!
```

Activity 1G (CLO2)

Activity Outcome: Identify programming errors in Java.
Identify logic errors in Java, implements programming style and documentation.

Procedures :

Step 1 : Open Notepad and type the following code:

```
/* Activity 1G */

class Act1G {
	public static void main(String[] args) {
		int mark = 59;
		if (mark < 50) // programmer mistake but syntaxly correct
			System.out.println("PASSED");
		else
			System.out.println("FAILED");
	}
}
```

Step 2 : Save the program as `Act1G.java` in the working directory, compile and run the program.
Step 3 : Observe the output.

```
nment\Lab Activity 1> javac .\Act1G.java
.\Act1G.java:5: error: unmappable character (0x9D) for encoding windows-1252
if (mark < 50) // programmer mistake but syntaxly correct System.out.println(ΓÇ£PASSEDΓÇ?);
                                                                              ^
.\Act1G.java:7: error: unmappable character (0x9D) for encoding windows-1252
System.out.println(ΓÇ£FAILEDΓÇ?);
                            ^
.\Act1G.java:6: error: 'else' without 'if'
else
^
3 errors
```

Step 4: Identify the logic error and correct the logic errors.

Edited Code:

```
/* Activity 1G */

class Act1G {
public static void main(String[] args) {
int mark = 59;
    if (mark > 50){ // programmer mistake but syntaxly correct
        System.out.println("PASSED");
    }else{
        System.out.println("FAILED");
}

}
}
```

Step 5 : Save your program, compile and run the program again.

Output:

```
nment\Lab Activity 1> java Act1G
PASSED
```