

Assignment 2

Snippet 1:

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

Error : Main method is not static in class Main, please define the main method as:

```
    public static void main(String[] args)
```

Correct Program:

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

Snippet 2:

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

Error: Main method not found in class Main, please define the main method as:

```
    public static void main(String[] args) or a JavaFX application class must extend  
    javafx.application.Application
```

Correct program :

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

Snippet 3:

```
public class Main {  
    public static int main(String[] args) {  
        System.out.println("Hello, World!");  
        return 0;  
    }  
}
```

Error: Main method must return a value of type void in class Main, please
define the main method as: public static void main(String[] args)

Correct Program :

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

Snippet 4:

```
public class Main {  
    public static void main() {
```

```
System.out.println("Hello, World!");  
  
}  
  
}
```

Error: Main method not found in class Main, please define the main method as:

public static void main(String[] args) or a JavaFX application class must extend
javafx.application.Application.

Correct program

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

Snippet 5:

```
public class Main {  
  
    public static void main(String[] args) {  
  
        System.out.println("Main method with String[] args");  
  
    }  
  
    public static void main(int[] args) {  
  
        System.out.println("Overloaded main method with int[] args");  
  
    }  
  
}
```

Error: In the second public static void main statement main method is written in lower case and it is a data type. It should be a string.

Main method is already defined in first main method, no need to write it again.

Correct program:

```
public class Main {  
    public static void main(String[] args) {  
        System.out.println("Main method with String[] args");  
  
        System.out.println("Overloaded main method with int[] args");  
    }  
}
```

Snippet 6:

```
public class Main {  
    public static void main(String[] args) {  
        int x = y + 10;  
        System.out.println(x);  
    }  
}
```

Error: cannot find symbol int x = y + 10; symbol: variable y
location: class Main

Correct program:

```
public class Main {  
    public static void main(String[] args) {  
        int y = 5;  
        int x = y + 10;  
        System.out.println(x);  
    }  
}
```

Snippet 7:

```
public class Main {  
    public static void main(String[] args) {  
        int x = "Hello";  
        System.out.println(x);  
    }  
}
```

Errorr:

incompatible types: String cannot be converted to int

```
int x = "Hello";
```

Correct program

```
public class Main{  
    public static void main(String[] args) {  
        int x = 2025;  
        System.out.println(x);  
    }  
}
```

Snippet 8:

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

Errorr: ')' expected System.out.println("Hello, World!"

Correct program:

```
public class Main2{  
    public static void main(String[] args) {  
        System.out.println("Hello, World!");  
    }  
}
```

Snippet 9:

```
public class Main {  
    public static void main(String[] args) {  
        int class = 10;  
        System.out.println(class);  
    }  
}
```

Error: not a statement

```
int class = 10;
```

^

Main2.java:4: error: ';' expected

```
int class = 10;
```

^

Main2.java:4: error: <identifier> expected

```
int class = 10;
```

^

Main2.java:5: error: <identifier> expectedSystem.out.println(class);

^Main.java:5: error: illegal start of type

```
System.out.println(class);
```

^Main.java:5: error: <identifier> expected System.out.println(class);

^Main.java:7: error: reached end of file while parsing

}

Correct program:

```
public class Main2 {  
    public static void main(String[] args) {  
        int a = 10;  
        System.out.println(a);  
    }  
}
```

Snippet 10:

```
public class Main {  
    public void display() {  
        System.out.println("No parameters");  
    }  
    public void display(int num) {  
        System.out.println("With parameter: " + num);  
    }  
    public static void main(String[] args) {  
        display();  
        display(5);  
    }  
}
```

Error: non-static method display() cannot be referenced from a static context

display();

Main.java:11: error: non-static method display(int) cannot be referenced from a static context

display(5);

Correct program: Method overloading not allowed in java

Snippet 11:

```
public class Main {  
    public static void main(String[] args) {  
        int[] arr = {1, 2, 3};  
        System.out.println(arr[5]);  
    }  
}
```

Error: Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 5 out of bounds for length 3 at Main.main(Main.java:5) array is not a data type we cannot declare array in integer.

Snippet 12:

```
public class Main {  
    public static void main(String[] args) {  
        while (true) {  
            System.out.println("Infinite Loop");  
        }  
    }  
}
```

Error: It will print infinite loop infinite times.

Solution: we have to add break statement after while loop

Snippet 13:

```
public class Main {  
    public static void main(String[] args) {  
        String str = null;  
        System.out.println(str.length());  
    }  
}
```

Exception: Exception in thread "main" java.lang.NullPointerException at
Main4.main(Main4.java:4)

Snippet 14:

```
public class Main {  
    public static void main(String[] args) {  
        double num = "Hello";  
        System.out.println(num);  
    }  
}
```

Error: incompatible types: String cannot be converted to double
double num = "Hello";

Solution: Java is a case sensitive language.

Snippet 15:

```
public class Main {  
    public static void main(String[] args) {  
        int num1 = 10;
```

```
double num2 = 5.5;
int result = num1 + num2;
System.out.println(result);
}
}
```

Error: incompatible types: possible lossy conversion from double to int

```
int result = num1 + num2;
```

Snippet 16:

```
public class Main {
    public static void main(String[] args) {
        int num = 10;
        double result = num / 4;
        System.out.println(result);
    }
}
```

Error: No error.

Output: 2.0

Snippet 17:

```
public class Main {
    public static void main(String[] args) {
        int a = 10;
        int b = 5;
        int result = a ** b;
        System.out.println(result);
    }
}
```

```
}  
}
```

Error: illegal start of expression int result = a ** b;

Correct program:

```
public class Main {  
    public static void main(String[] args) {  
        int a = 10;  
        int b = 5;  
        int result = a * b;  
        System.out.println(result);  
    }  
}
```

Snippet 18:

```
public class Main {  
    public static void main(String[] args) {  
        int a = 10;  
        int b = 5;  
        int result = a + b * 2;  
        System.out.println(result);  
    }  
}
```

Output is 20, but output should be 30. So to solve this we have to add parenthesis () for a+b then it will give a correct output.

Snippet 19:

```

public class Main {
    public static void main(String[] args) {
        int a = 10;
        int b = 0;
        int result = a / b;
        System.out.println(result);
    }
}

```

Exception: Exception in thread "main" java.lang.ArithmeticException: / by zero
 at Main.main(Main.java:6)

we can not divide any number by zero because in java 0 and 1 are used to represent binary representation of decimals.

Snippet 20:

```

public class Main {
    public static void main(String[] args) {
        System.out.println("Hello, World")
    }
}

```

Error: ';' expected

```

System.out.println("Hello, World")

```

Correct program

```

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

```

```
}  
}
```

Snippet 21:

```
public class Main {  
    public static void main(String[] args) {  
        System.out.println("Hello, World!");  
        // Missing closing brace here  
    }  
    Erroor : reached end of file while parsing  
    }.
```

Snippet 22:

```
public class Main {  
    public static void main(String[] args) {  
        static void displayMessage() {  
            System.out.println("Message");  
        }  
    }  
}  
Errorr: error: illegal start of expression  
    static void displayMessage() {  
Main.java:8: error: class, interface, or enum expected  
    }
```

Snippet 23:

```
public class Confusion {  
    public static void main(String[] args) {  
        int value = 2;  
        switch(value) {  
            case 1:  
                System.out.println("Value is 1");  
            case 2:  
                System.out.println("Value is 2");  
            case 3:  
                System.out.println("Value is 3");  
            default:  
                System.out.println("Default case");  
        }  
    }  
}
```

Output: Value is 2

Value is 3

Default case

Solution: To prevent print of default value we can add break statement in case 3.

Snippet 24:

```
public class MissingBreakCase {
```

```

public static void main(String[] args) {
    int level = 1;
    switch(level) {
    case 1:
        System.out.println("Level 1");
    case 2:
        System.out.println("Level 2");
    case 3:
        System.out.println("Level 3");
    default:
        System.out.println("Unknown level");
    }
}

```

Output: Level 1

Level 2

Level 3

Unknown level

Solution:

To print only level 1 we need to add break statement in case 1 because we are using while statement so in this break is mandatory after every case. Other wisw it will print all case output tiil it gets break .

Snippet 25:

```

public class Switch {

```

```

public static void main(String[] args) {
    double score = 85.0;
    switch(score) {
    case 100:
        System.out.println("Perfect score!");
        break;
    case 85:
        System.out.println("Great job!");
        break;
    default:
        System.out.println("Keep trying!");
    }
}

```

Error: witch.java:5: error: incompatible types: possible lossy conversion from double to int

```

switch(score) {

```

Snippet 26:

```

public class Switch {
    public static void main(String[] args) {
        int number = 5;
        switch(number) {
        case 5:
            System.out.println("Number is 5");
            break;
        case 5:
            System.out.println("This is another case 5");

```



```
break;
```

```
default:
```

```
System.out.println("This is the default case");
```

```
}
```

```
}
```

```
}
```

Solution we can not give same number for all cases . it will show duplicate label error.