Section 1: Java Data Types

1. What are the different primitive data types available in Java?

Ans: Java has 8 primitive datatypes those are used to data like numbers, characters and true/false values.

1. int
2. byte
3. short
4. long
5. float
6. double
7. char
8. Boolean

2. Explain the difference between primitive and non-primitive data types in Java?

Ans:

Primitive Datatype	Non-Primitive Datatype
1. Fixed memory	1. Not Fixed memory
2. Used for simple operations	2.Used for complex operations
3. Predefined by Java	3.Created by programmers
4.Stored in stack	4.Stored in heap
Ex: int, char	Ex: String, Array

3. Write a Java program that demonstrates the use of all primitive data types?

Ans:

Program:

```
package Datatypes;

public class Test {

	public static void main(String[] args) {

		byte b = 100;

		short s = 2000;

		int i = 100000;

		long l = 123456789L;

		float f = 5.75f;

		double d = 19.99;

		char c = 'A';
```

```
        boolean bool = true;
        System.out.println(b);
        System.out.println(s);
        System.out.println(i);
        System.out.println(l);
        System.out.println(f);
        System.out.println(d);
        System.out.println(c);
        System.out.println(bool);
    }
}
```

Output :

100

2000

100000

123456789

5.75

19.99

A

true

4. What is type casting? Provide an example of implicit and explicit casting in Java?

Ans: Typecasting means converting one datatype to another datatype. There are two types

1. Widening : Converting small datatype to large datatype which done by automatically.

Ex: int -> long -> float

2. Narrowing : Converting larger datatype to smaller datatype which done by manually.

Ex: float -> long -> int

Program :

```
package Type_Casting;
public class Casting {
```

```
    public static void main(String[] args) {
        int num = 10;
        System.out.println("value of num : " + num);
        double d = num;
        System.out.println("value of d : " + d);
        // Narrowing
        int a  = (int) d;
        System.out.println("value of a : " + a);
    }
}
```

Output :

value of num : 10

value of d : 10.0

value of a : 10

5. What is the default value of each primitive data type in Java?

Ans: When a primitive type is declared as an instance variable, Java assigns a default value if not initialized.

1. byte – 0
2. short – 0
3. int – 0
4. long – 0L
5. float – 0.0f
6. double – 0.0d
7. char – \u0000
8. boolean - false

Section 2 : Java Control Statements

1. What are control statements in Java? List the types with examples ?

Ans: Control statements are used to control the flow of execution of the program.

1. Conditional Statements : if, if-else, if-else-if(ladder), nested if, switch
2. Looping Statements: while, do-while, for
3. Jumping statements : break, continue, return

Condtional statements :

1. If :

int num1 = 20;

```
if (num1 > 15) {
   System.out.println("Number is greater than 15");
}
```

2. If-else :

```
int num2 = 7;
if (num2 > 10) {
   System.out.println("Greater");
} else {
   System.out.println("Smaller");
}
```

3. if-else-if ladder :

```
int marks = 68;
if (marks >= 90) {
   System.out.println("Grade A");
} else if (marks >= 75) {
   System.out.println("Grade B");
} else {
   System.out.println("Grade C");
}
```

4. switch :

```
int day = 5;
switch(day) {
   case 1:
      System.out.println("Monday");
      break;
   case 2:
      System.out.println("Tuesday");
      break;
   case 3:
```

```
        System.out.println("Wednesday");
        break;
    default:
        System.out.println("Invalid day");
}
```

Looping statements :

1. for loop :

```
for (int i = 10; i <= 15; i++) {
    System.out.println(i);
}
```

2. While loop :

```
int j = 3;
while (j <= 7) {
    System.out.println(j);
    j++;
}
```

3. do-while loop :

```
int k = 8;
do {
    System.out.println(k);
    k++;
} while (k <= 12);
```

Jumping statements :

1. break :

```
for (int m = 1; m <= 5; m++) {
    if (m == 4)
        break;
    System.out.println(m);
}
```

2. continue :

```
for (int n = 1; n <= 5; n++) {
   if (n == 2)
      continue;
   System.out.println(n);
}
```

3. return :

```
public static int add(int a, int b) {
   return a + b;
}
```

2. Write a Java program to demonstrate the use of if-else and switch-case statements?

Program :

```
package Control_Statements;
public class ControlStatementsDemo {
   public static void main(String[] args) {
         int num = 18;
         if (num % 2 == 0) {
               System.out.println(num + " is even.");
         } else {
               System.out.println(num + " is odd.");
         }
         int val = 2;
         switch(val) {
               case 1: System.out.println("Tea");
               break;
               case 2: System.out.println("Coffee");
               break;
               case 3: System.out.println("Juice");
```

```
                break;
                default: System.out.println("Invalid Input ");
            }
        }
}
```

Output :

18 is even

Coffee

3. What is the difference between break and continue statements?

Ans:

Break	Continue
1.Immediately terminates the loop	1. Used to skip the current iteration
2.Ends the loop completely	2.Keeps the loop running
3.Used in switch	3.Not used in switch

4. Write a Java program to print even numbers between 1 to 50 using a for loop?

Ans:

Program:

```
public class EvenNumbers {
    public static void main(String[] args) {
        for (int i = 1; i <= 50; i++) {
            if (i % 2 == 0) {
                System.out.print(i + " ");
            }
        }
    }
}
```

Output: 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50

5. . Explain the differences between while and do-while loops with examples?

Ans:

While	Do-While
1.Condition is checked before executing	1. Condition is checked after executing
2.It will not execute if condition is false	2. Atleast once will execute even condition is wrong.

Program:

```
public class Test {
public static void main(String[] args) {
        int a = 1;
        while (a < 5) {
                System.out.println(a+ " ");
                a++;
         }
        int b = 1;
        do {
                System.out.print(b+ " ");
                b++;
                } while (b < 5);
        }
}
```

Output: 1 2 3 4 1 2 3 4

Section 3 : Java Keywords and Operators

1. What are keywords in Java? List 10 commonly used keywords?

Ans: Java keywords are reserved words that have predefined meanings in Java.

ex : if, else, for, while, try, throw, throws, catch, int, char, float, static, public, private,

2. Explain the purpose of the following keywords: static, final, this, super?

Ans:

Static : 1.It is belongs to class, not to instance

2. shared across all objects of class

3. useds for variable, methods

Final : 1.used to create constants

2. used for variables, methods

This : 1. Refers to the current object

2. used to avoid name conflicts when there is same name.

Super: 1. Refers to parent class

2. used to access parent class variables and methods

3. What are the types of operators in Java?

Ans: Operators are symbols used to perform operations on variables and values.

1. Arithmetic Operators
2. Relational Operators
3. Logical Operators
4. Assignment Operators
5. Unary Operators
6. Bitwise Operators
7. Shift Operators
8. instanceOf Operators

4. Write a Java program demonstrating the use of arithmetic, relational, and logical operators?

Ans :

Program :

```
public class Test {
public static void main(String[] args) {
    int a = 10, b = 20;
    // Arithmetic
    System.out.println(a + b);
    // Relational
    System.out.println(a > b);
    // Logical
    System.out.println(a > 5 && b > 15);
  }
}
```

Output : 30

false

true

5. What is operator precedence? How does it affect the outcome of expressions?

Ans : Operator precedence is priority of operators in expression. It decides which operation should execure first when multiple operators are in same expression

ex :
```
public class Test {
public static void main(String[] args) {
    int result = 10 + 5 * 2;
    System.out.println("result :  " +result);
  }
}
```
Output : 20

Java DataTypes :

6. What is the size and range of each primitive data type in Java?

Ans :

byte: 1byte - 8 bits, values from -128 to 127

short: 2 bytes - 16 bits, values from -32,768 to 32,767

int: 4 bytes - 32 bits, values from -2^{31} to $2^{31}-1$

long: 4 bytes - 64 bits, values from -2^{63} to $2^{63}-1$

float: 2 bytes - 32 bits, stores fractional numbers approx. ±3.4e+38

double: 4 bytes - 64 bits, stores fractional numbers approx. ±1.7e+308

char: 2 bytes - 16 bits, stores a single Unicode character from '\u0000' to '\uffff' (0 to 65,535)

boolean**:** size not precisely defined, values are true or false

7. How does Java handle overflow and underflow with numeric types?

Ans : Java does not raise errors on overflow and underflow, instead it wraps the values according to 2's compliment.

8. Write a program to convert a double value to an int without data loss.

```
public class Convert {
  public static void main(String[] args) {
```

```
        double d = 123.0;
        int i = (int) d;
        System.out.println(i);
    }
}
```

9. What is the difference between char and String in Java?

Ans : 1. Char is a primitive and it stores only single value.

2. String is Object and it is a sequence of characters

10. Explain wrapper classes and their use in Java.

Ans : Wrapper classes are used to convert primitives to objects to work with collections.

ex: int – Integer

Java Control Statements:

6. Write a Java program using nested if statements.

Ans:

```
public class Test {
    public static void main(String[] args) {
        int age = 20;
        if (age >= 18) {
            if (age < 60) {
                System.out.println("Adult");
            } else {
                System.out.println("Senior");
            }
        } else {
            System.out.println("Minor");
        }
    }
}
```

7. Write a Java program to display the multiplication table of a number using a loop.

Ans:

```
public class Table {
```

```
    public static void main(String[] args) {
        int num = 7;
        for (int i = 1; i <= 10; i++) {
            System.out.println(num + " * " + i + " = " + (num * i));
        }
    }
}
```

8. How do you exit from nested loops in Java?

Ans: By using break statement we can exit from nested loops.

9. Compare and contrast for, while, and do-while loops.

Ans: 1. for – for loop can be used when we know the no of iterations

2. while – while loop is used to execute the loop till the condition is true.

3. do-while – it is used when we want execute the loop at least once.

10. Write a program that uses a switch-case to simulate a basic calculator.

Ans :

```
public class CalculatorSwitch {
    public static void main(String[] args) {
        int a = 5;
        int b = 5;
        char operator = '+';

        double result;

        switch (operator) {
            case '+':
                result = a + b;
                System.out.println(a + " + " + b + " = " + result);
                break;
```

```
            case '-':
                result = a - b;
                System.out.println(a + " - " + b + " = " + result);
                break;

            case '*':
                result = a * b;
                System.out.println(a + " * " + b + " = " + result);
                break;

            case '/':
                result = (double) a / b;
                System.out.println(a + " / " + b + " = " + result);
                break;

            default:
                System.out.println("Invalid operator.");
        }
    }
}
```

Java Keywords and Operators :

6. What is the use of the `instanceof` keyword in Java?

Ans : instanceOf is used to check if an object is an instance of class or not .

7. Explain the difference between `==` and `.equals()` in Java.

Ans : 1. ‘==’ is used to compare references

2. equals() is used to compare values/data.

8. Write a program using the ternary operator.

Ans : public class Test {

```
    public static void main(String[] args) {
        int a = 10, b = 20;
        int max = (a > b) ? a : b;
        System.out.println("Max = " + max);
    }
}
```

9. What is the use of `this` and `super` in method overriding?

Ans : this keyword refers to current class.

Super keyword refers to the immediate parent.

10. Explain bitwise operators with examples.

Ans : & (bitwise AND)

| (bitwise OR)

^ (bitwise XOR)

~ (bitwise NOT)

<< (left shift)

>> (right shift)

Program :

```
public class Test {
    public static void main(String[] args) {
        int x = 12;
        int y = 5;

        System.out.println(x & y);
        System.out.println(x | y);
        System.out.println(x ^ y);
        System.out.println(~x);
        System.out.println(x << 2);
        System.out.println(x >> 2);
        System.out.println(x >>> 2);
```

```
    }
}
```