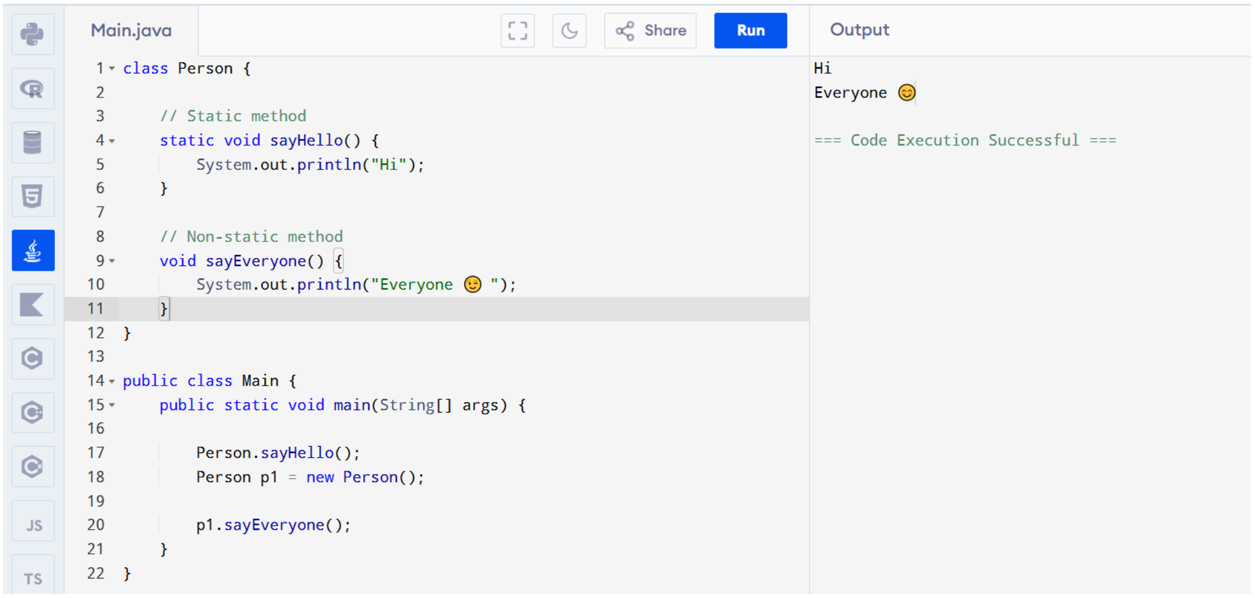
EUROPEAN UNIVERSITY OF BANGLADESH

Assignment	: 01
Course name	: Object Oriented Programming Sessional
Course code	: CSE-212
Submitted to	**:**
Name	: Sabrin Afroz
Designation	: Lecturer

Submitted by	**:**
Name	: MD Sayem Hossen
ID	: 250221033
Semister	: 2nd
Date	: 19 - 11- 2025

1, Static Class

```
class Person {

   // Static method
   static void sayHello() {
      System.out.println("Hi");
   }

   // Non-static method
   void sayEveryone() {
        System.out.println("Everyone 😉 ");
    }
}

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

        Person.sayHello();
        Person p1 = new Person();

        p1.sayEveryone();
    }
}
```

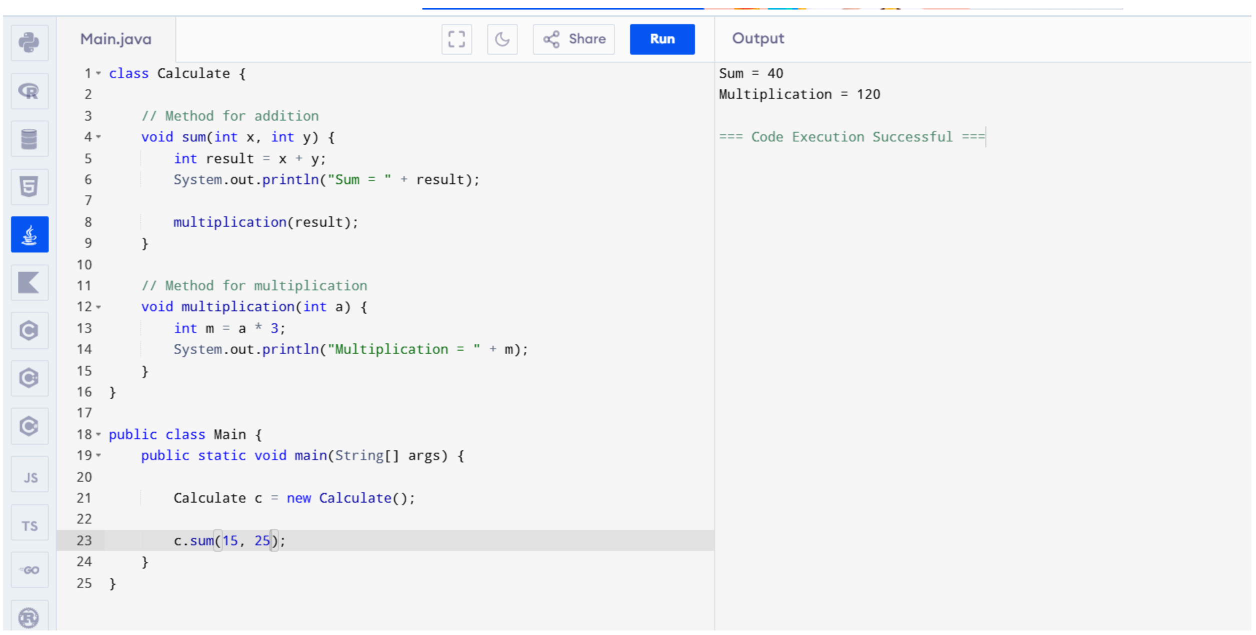


```
class Person {

    // Static method
    static void sayHello() {
        System.out.println("Hi");
    }

    // Non-static method
    void sayEveryone() {
        System.out.println("Everyone 😉 ");
    }
}

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

        Person.sayHello();
        Person p1 = new Person();

        p1.sayEveryone();
    }
}
```

```
Hi
Everyone 😉

=== Code Execution Successful ===
```

2, Static Calculations

```
class Calculate {

    // Method for addition
    void sum(int x, int y) {
        int result = x + y;
        System.out.println("Sum = " + result);

        multiplication(result);
    }

    // Method for multiplication
    void multiplication(int a) {
        int m = a * 3;
        System.out.println("Multiplication = " + m);
    }
}

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

        Calculate c = new Calculate();

        c.sum(15, 25);
    }
}
```

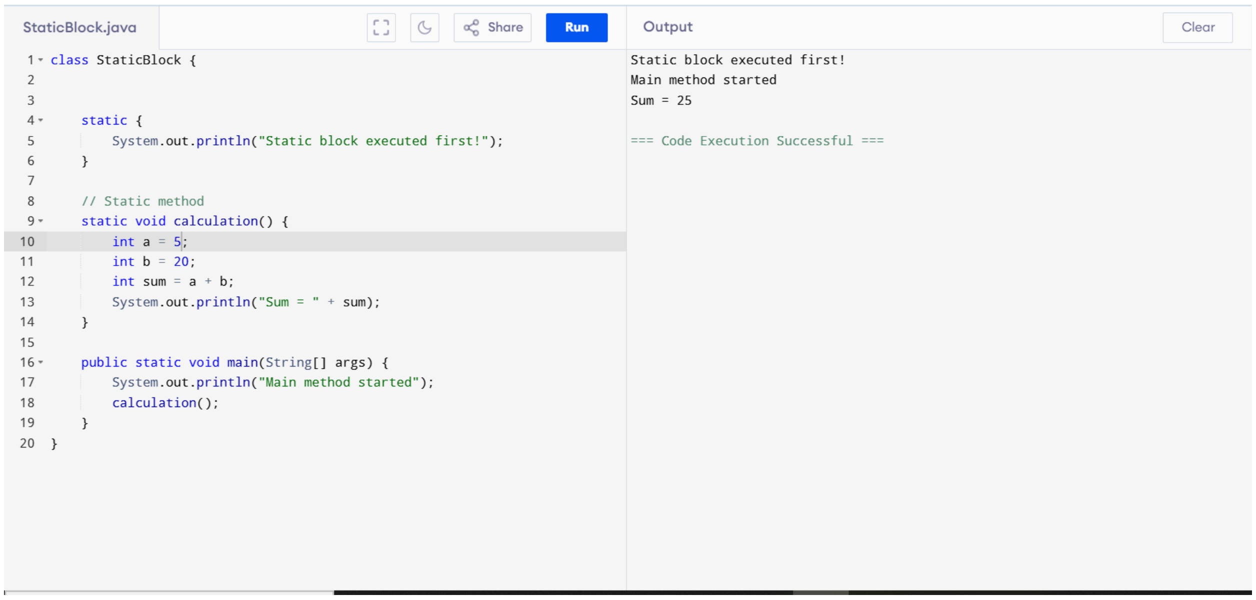


```
class Calculate {

    // Method for addition
    void sum(int x, int y) {
        int result = x + y;
        System.out.println("Sum = " + result);

        multiplication(result);
    }

    // Method for multiplication
    void multiplication(int a) {
        int m = a * 3;
        System.out.println("Multiplication = " + m);
    }
}

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

        Calculate c = new Calculate();

        c.sum(15, 25);
    }
}
```

```
Sum = 40
Multiplication = 120

=== Code Execution Successful ===
```

3, Static Block

```
class StaticBlock {


    static {
        System.out.println("Static block executed first!");
    }

    // Static method
    static void calculation() {
        int a = 5;
        int b = 20;
        int sum = a + b;
        System.out.println("Sum = " + sum);
    }

    public static void main(String[] args) {
        System.out.println("Main method started");
        calculation();
    }
}
```



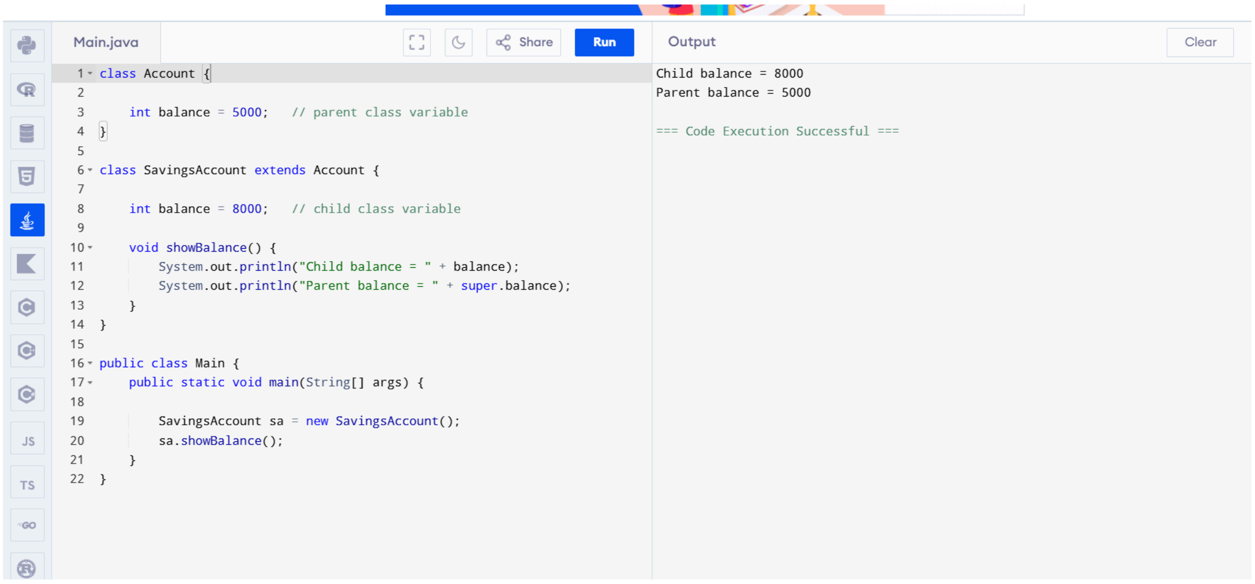
```
class StaticBlock {


    static {
        System.out.println("Static block executed first!");
    }

    // Static method
    static void calculation() {
        int a = 5;
        int b = 20;
        int sum = a + b;
        System.out.println("Sum = " + sum);
    }

    public static void main(String[] args) {
        System.out.println("Main method started");
        calculation();
    }
}
```

```
Static block executed first!
Main method started
Sum = 25

=== Code Execution Successful ===
```

4, Super Keyword Variable Call

```
class Account {

    int balance = 5000;   // parent class variable
}

class SavingsAccount extends Account {

    int balance = 8000;   // child class variable

    void showBalance() {
        System.out.println("Child balance = " + balance);
        System.out.println("Parent balance = " + super.balance);
    }
}
```

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

        SavingsAccount sa = new SavingsAccount();
        sa.showBalance();
    }
}
```



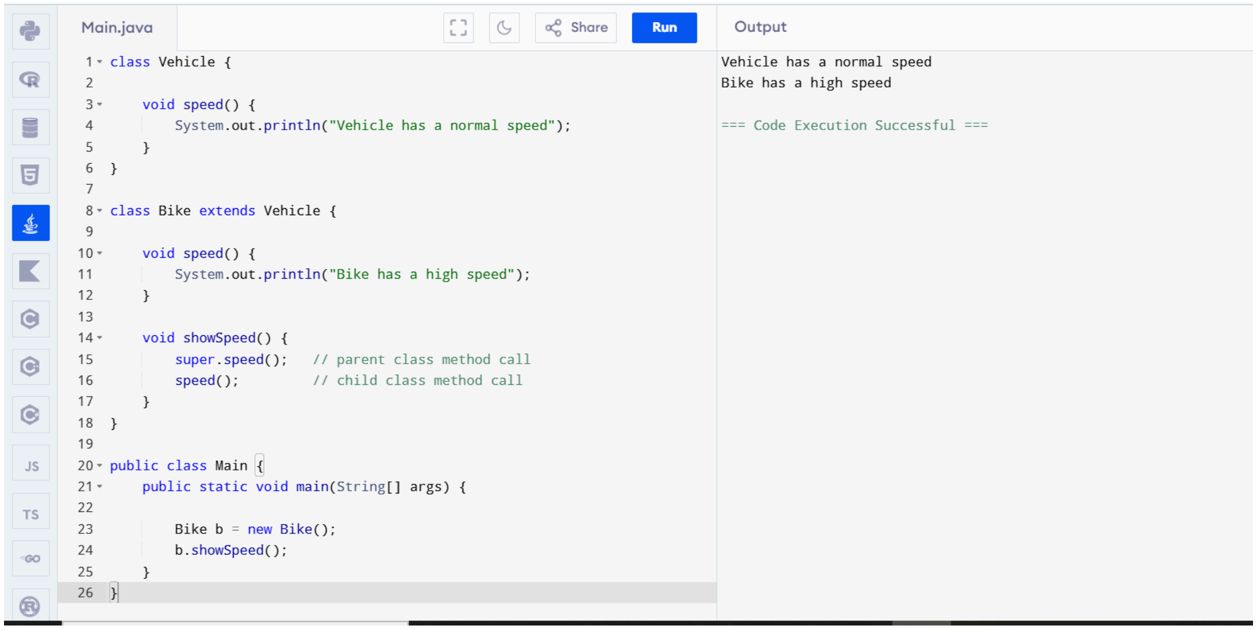
```
class Account {

    int balance = 5000;   // parent class variable
}

class SavingsAccount extends Account {

    int balance = 8000;   // child class variable

    void showBalance() {
        System.out.println("Child balance = " + balance);
        System.out.println("Parent balance = " + super.balance);
    }
}

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

        SavingsAccount sa = new SavingsAccount();
        sa.showBalance();
    }
}
```

```
Child balance = 8000
Parent balance = 5000

=== Code Execution Successful ===
```

5, Super Keyword Method

```
class Vehicle {

    void speed() {
        System.out.println("Vehicle has a normal speed");
    }
}

class Bike extends Vehicle {

    void speed() {
        System.out.println("Bike has a high speed");
    }

    void showSpeed() {
        super.speed();   // parent class method call
        speed();         // child class method call
    }
}
```

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

        Bike b = new Bike();
        b.showSpeed();
    }
}
```

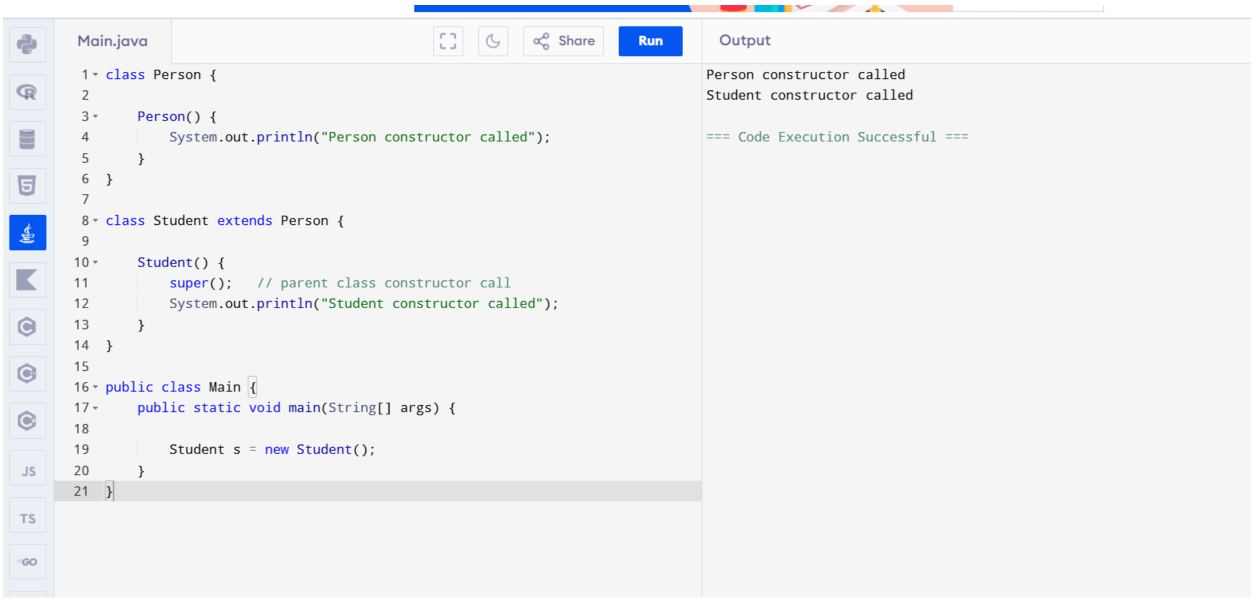


```
class Vehicle {

    void speed() {
        System.out.println("Vehicle has a normal speed");
    }
}

class Bike extends Vehicle {

    void speed() {
        System.out.println("Bike has a high speed");
    }

    void showSpeed() {
        super.speed();    // parent class method call
        speed();          // child class method call
    }
}

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

        Bike b = new Bike();
        b.showSpeed();
    }
}
```

```
Vehicle has a normal speed
Bike has a high speed

=== Code Execution Successful ===
```

6, Super Keyword Constructor

```
class Person {

    Person() {
        System.out.println("Person constructor called");
    }
}

class Student extends Person {

    Student() {
        super();   // parent class constructor call
        System.out.println("Student constructor called");
    }
}

public class Main {
    public static void main(String[] args) {
        Student s = new Student();
    }
}
```

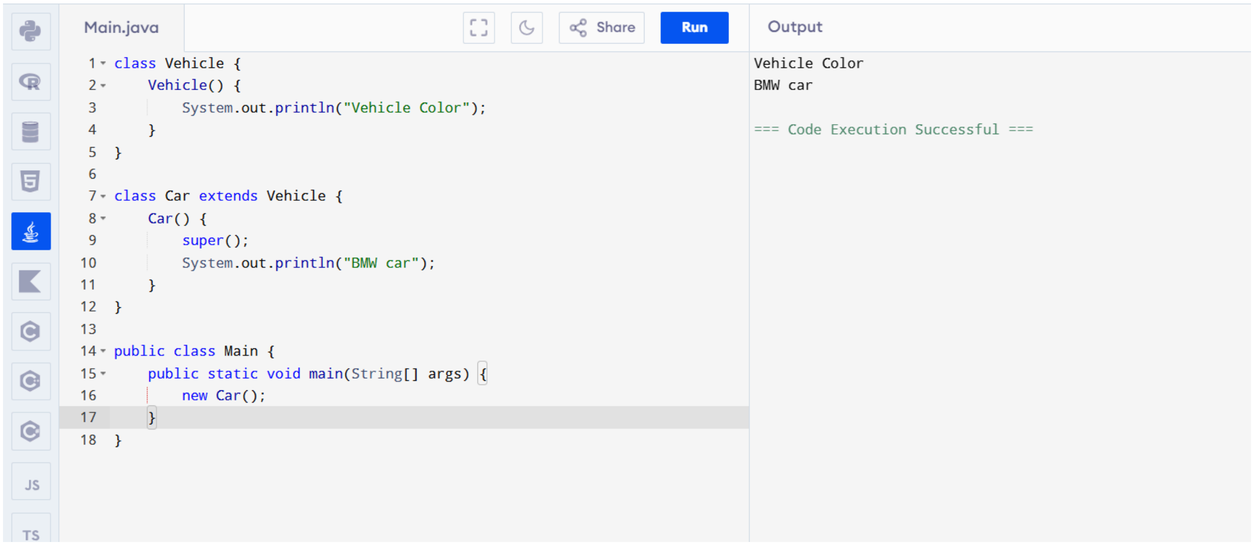


```
class Person {

    Person() {
        System.out.println("Person constructor called");
    }
}

class Student extends Person {

    Student() {
        super();   // parent class constructor call
        System.out.println("Student constructor called");
    }
}

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

        Student s = new Student();
    }
}
```

```
Person constructor called
Student constructor called

=== Code Execution Successful ===
```

7, Vehicle and Car Programme

```
class Vehicle {
    Vehicle() {
        System.out.println("Vehicle Color");
    }
}

class Car extends Vehicle {
    Car() {
        super();
        System.out.println("BMW car");
    }
}

public class Main {
    public static void main(String[] args) {
        new Car();
    }
}
```



```java
class Vehicle {
    Vehicle() {
        System.out.println("Vehicle Color");
    }
}

class Car extends Vehicle {
    Car() {
        super();
        System.out.println("BMW car");
    }
}

public class Main {
    public static void main(String[] args) {
        new Car();
    }
}
```

```
Vehicle Color
BMW car

=== Code Execution Successful ===
```

– THE END