q-1 Write a program to demonstrate Tightly Coupled code.
Bike_Tight Class

```
package tight_coupling;
public class bike_tight {
    public void move()
    {
        System.out.println("I am going by bike");
    }
}
```

Car_Tight Class

```
package tight_coupling;
public class car_tight {
    public void move()
    {
        System.out.println("I am going by car");
    }
}
```

Plane_Tight Class

```
package tight_coupling;
public class plane_tight {
    public void move()
    {
        System.out.println("I am going by plane");
    }
}
```

Traveller_Tight Class

```
package tight_coupling;
public class traveller_tight {
    plane_tight p = new plane_tight();
    public void start_journey() {
        p.move();
    }
}
```

Tight _Coupling Class

```
package tight_coupling;
public class tight_coupling {
    public static void main(String[] args) {
        traveller_tight t=new traveller_tight();
        t.start_journey();
    }
}
```

Output

```
/snap/intellij-idea-community/208/jbr/bin/java -javaagent:/snap/intellij-idea-community/208/lib/idea_rt.jar=42189:/snap/intellij-idea-community/208/bin -Dfile
  .encoding=UTF-8 -classpath "/home/siddharth/assignment/spring assignment-1/out/production/spring assignment-1" tight_coupling.tight_coupling
I am going by plane

Process finished with exit code 0
```

q-2 Write a program to demonstrate Loosely Coupled code.

Interface Vehicle

```
package loose_coupling;
public interface vehicle {
    void move();
}
```

Bike_Loose Class

```
package loose_coupling;
public class bike_loose implements  vehicle {
    @Override
    public void move() {
        System.out.println("I am on a bike");
    }
}
```

Car_Loose Class

```
package loose_coupling;
public class car_loose implements  vehicle {
    @Override
    public void move() {
        System.out.println("I am on a car");
    }
}
```

Plane_Loose Class

```
package loose_coupling;
public class plane_loose implements  vehicle {
    @Override
    public void move() {
        System.out.println("I am on a plane");
    }
}
```

Traveller_Loose Class

```
package loose_coupling;
public class traveller_loose {
    vehicle v;
    public vehicle getV() {
        return v;
    }
    public void setV(vehicle v) {
        this.v = v;
    }
    public void  start_journey()
    {
        v.move();
    }
}
```

Coupling_Loose Class

```
package loose_coupling;
public class coupling_loose {
    public static void main(String[] args) {
        vehicle v=new plane_loose();
        traveller_loose t=new traveller_loose();
        t.setV(v);
        t.start_journey();
    }
```

```
}
```

Output

```
/snap/intellij-idea-community/208/jbr/bin/java -javaagent:/snap/intellij-idea-community/208/lib/idea_rt.jar=44491:/snap/intellij-idea-community/208/bin -Dfile
  .encoding=UTF-8 -classpath "/home/siddharth/assignment/spring assignment-1/out/production/spring assignment-1" loose_coupling.coupling_loose
I am on a plane

Process finished with exit code 0
```

q-3 Use @Compenent and @Autowired annotations to in Loosely Coupled code for dependency management
q-4 Get a Spring Bean from application context and display its properties.
q-5 Demonstrate how you will resolve ambiguity while autowiring bean (Hint : @Primary)
q-6 Perform Constructor Injection in a Spring Bean

Interface Vehicle

```
package com.example.demo;
public interface vehicle {
        void move();
}
```

Bike_Loose Class

```
package com.example.demo;
import org.springframework.context.annotation.Primary;
import org.springframework.stereotype.Component;
@Component
@Primary
public class bike_loose implements  vehicle {
    @Override
    public void move() {
        System.out.println("I am on a bike");
    }
}
```

Car_Loose Class

```
package com.example.demo;
import org.springframework.stereotype.Component;
@Component
public class car_loose implements  vehicle {
    @Override
    public void move() {
        System.out.println("I am in a car");
    }
}
```

Plane_Loose Class

```
package com.example.demo;
import org.springframework.stereotype.Component;
@Component
public class plane_loose implements  vehicle {
    @Override
    public void move() {
        System.out.println("I am on a plane");
    }
}
```

Traveller_Loose Class

```
package com.example.demo;
import org.springframework.beans.factory.annotation.Autowired;
```

```java
import org.springframework.stereotype.Component;
@Component
public class traveller_loose {
    int price=1000;
    @Autowired
    vehicle v;
   public void setV(vehicle v) {
        this.v = v;
    }
    public void  start_journey()
    {
        v.move();
    }
}
```

DemoApplication Class

```java
package com.example.demo;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.context.ApplicationContext;
@SpringBootApplication
public class DemoApplication {
   public static void main(String[] args) {
      ApplicationContext q3=SpringApplication.run(DemoApplication.class, args);
      traveller_loose traveller=q3.getBean(traveller_loose.class);
      traveller.start_journey();
      System.out.println(traveller.price);
   }
}
```