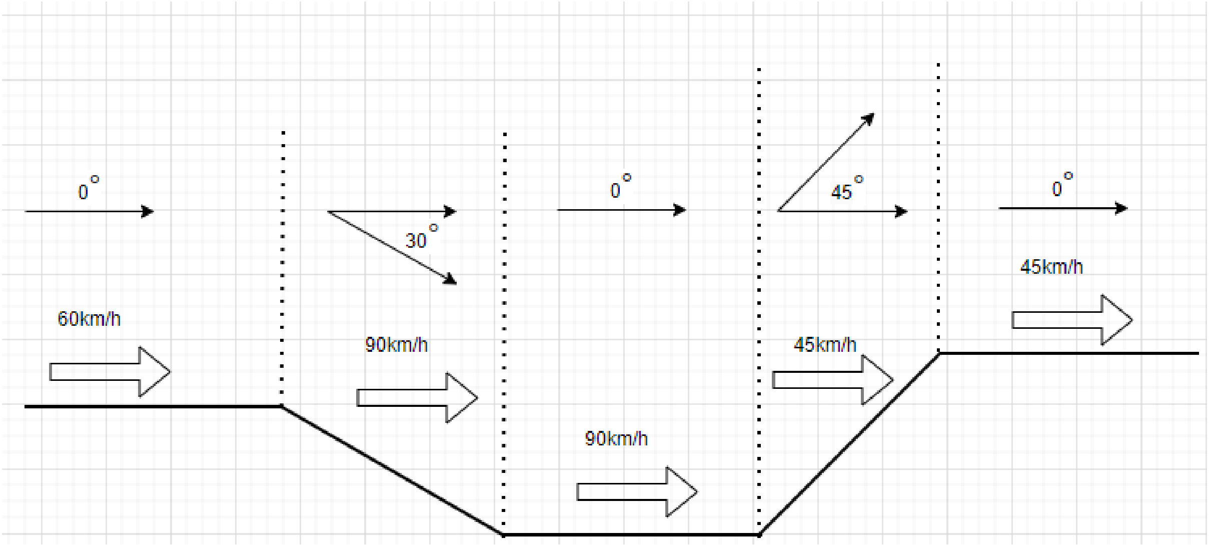
Question 1:
Create a method that will compute the final speed given a map and a starting speed.
The speed will increase of decrease proportionally to the inline of the land as shown in the image below:



Initial Code:

```
public class GamePlatform {
    public static double getFinalSpeed(double initialSpeed, int[] inclinations) {
        throw new UnsupportedOperationException("Waiting to be implemented.");
    }

    public static void main(String[] args){
        System.out.println(getFinalSpeed(60.0, new int[] { 0, -30, 0, 45, 0 })); //should print 45
    }
}
```

Question 2:
Create a method to return all combinations of one colour and one shape.

Initial Code:

```
import java.util.List;

public class Drawing {
    public String[] colours;
    public String[] shapes;

    public static class Art {
        public String colour;
        public String shape;

        public Art(String colour, String shape) {
            this.colour = colour;
            this.shape = shape;
        }
    }

    public IceCreamMachine(String[] colours, String[] shapes) {
        this.colours = colours;
        this.shapes = shapes;
    }

    public List<Art > mix() {
        throw new UnsupportedOperationException("Waiting to be implemented");
    }

    public static void main(String[] args) {
        Drawing draw = new Drawing (new String[]{
                "red", "blue"
        }, new String[]{
                "circle"
        });
        List<Art> mixes= draw.mix();

        /*
         * Should print:
         * red, circle
         * blue, circle
         */
        for (Art art: mixes) {
            System.out.println(art.colour + ", " + art.shape);
        }
    }
}
```

Question 3:
Given a studentRepository class is used in a Spring Boot web application.
Complete the findStudentByName method to return all students that start with the nameStartWith string.The return list should contain Student Object with the name,gender.

Following class provided below is used to Configure Spring:

```
@Configuration
@Import(StudentRepository.java)
class Config {
        @Bean
        public DriverManagerDatasource dataSource() {
            DriverManagerDataSource ds = new DriverManagerDataSource();
            ds.setDriverClassName(“org.h2.Driver”);
            ds.setUrl(“jdbc:h2:mem:test;DB_CLOSE_DELAY=-1”);
            return ds;
        }

        @Bean
        public JdbcTemplate jdbcTemplate(DriverManagerDatasource  ds) {
            return new JdbcTemplate(ds);
        }
}
```

Initial Code:

```
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.jdbc.core.*;
import org.springframework.stereotype.Repository;
import javax.annotation.PostConstruct;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.*;

class Student{
    public String name;
    public String gender;

    public Student(String name, String gender) {
        this.name = name;
        this.gender = gender;
    }
}

@Repository
public class StudentRepository {

    @Autowired
    private JdbcTemplate template;

    @PostConstruct
    public void createTable() {
        template.execute("CREATE TABLE student(id bigint auto_increment primary key,
name VARCHAR(50), gender VARCHAR(1))");
    }

    public void createStudent(String name, String gender) {
        template.update("INSERT INTO student(id, name, gender) VALUES (?,?,?)",
                        null, name, gender);
    }

    public List<Student> findStudentByName(String nameStartsWith) {
        // Write your code here
        return null;
    }
}
```

Question 4:
If you use the same **PreAuthorize** configuration on each method in a class, how can you make the annotations cleaner?

Question 5:
Given tables below:

Table: players	
Player_no	Name
1	Alan
2	James
3	Alan
4	Bob

Table: scores		
Player_no	Sport	Score
3	Chess	10
3	Swimming	20
2	Swimming	10
1	Swimming	5

Write Sql to show result below:

Name	Sport	Total Score
Alan	Swimming	20
James	Swimming	10
Alan	Chess	10
Alan	Swimming	5