**תרגיל מספר 3**

**Interfaces**

**תרגיל 1 - פרויקט Shape**

1. יש להוסיף לפרויקט של Shape את ה-interface Movable בקובץ נוסף, אשר ידרוש מכל המחלקות בפרויקט לממש את רשימת המטודות הבאות:

void moveUp( )

void moveDown( )

void moveLeft( )

void moveRigth( )

1. יש ליצור interface GeometricableObject אשר מכיל את המטודות ()perimeter ו- area() ולשנות את הקוד כך שכל הצורות יממשו אותן. (יש לשים לב מה ניתן לעשות עם Line (!!
2. יש להוסיף מטודה areaOfAll ב-ShapeContainer המקבלת מערך של אוביקטים אשר מימשו את GeometricableObject ומחזירה את סכום השטחים של כל הצורות יחד.
3. יש להגדיר מערך של 3 צורות ב- main אשר ניתן יהיה להזיז את כולם ביחד לכוון מסויים לפי בקשת המשתמש.

**תרגיל 2 - פרויקט Queue**

1. יש לממש interface Queueable המגדיר מימוש FIFO ומכיל את המטודות הבאות:

* void add(int value)
* int remove()
* boolean isEmpty()

1. הגדר class ArrayQueue המממש את ה-interface באופן הבא:

|  |  |
| --- | --- |
| 1. **Constructor Summary** | |
| [**ArrayQueue**](http://www.home.hs-karlsruhe.de/~pach0003/informatik_1/aufgaben/en/doc/de/hska/java/exercises/interfaces/ArrayQueue.html#ArrayQueue())()            creates a new queue with the initial capacity of 100 elements |  |
| [**ArrayQueue**](http://www.home.hs-karlsruhe.de/~pach0003/informatik_1/aufgaben/en/doc/de/hska/java/exercises/interfaces/ArrayQueue.html#ArrayQueue(int))(int capacity)            creates a new queue with the inital capacity |  |

|  |  |
| --- | --- |
| **Method Summary** | |
| void | [**add**](http://www.home.hs-karlsruhe.de/~pach0003/informatik_1/aufgaben/en/doc/de/hska/java/exercises/interfaces/ArrayQueue.html#add(int))(int element)            Adds the given value to this queue. |
| boolean | [**isEmpty**](http://www.home.hs-karlsruhe.de/~pach0003/informatik_1/aufgaben/en/doc/de/hska/java/exercises/interfaces/ArrayQueue.html#isEmpty())()            Returns true if this queue contains one value at least. |
| int | [**remove**](http://www.home.hs-karlsruhe.de/~pach0003/informatik_1/aufgaben/en/doc/de/hska/java/exercises/interfaces/ArrayQueue.html#remove())()            Removes and returns the value of the element that has be added to this queue at first. |

1. הגדר main שבו אובייקט מסוג ArrayQueue והפעל את המטודות.

**תרגיל 3 -Strategy**

1. בפרויקט חדש, יש להגדיר את:

**public** **interface** Strategy {

**boolean** checkTemperature(**int** temperature);

}

1. יש להוסיף 2 מחלקות המממשות את הinterface הנקראות:

**public** **class** HikeStrategy **implements** Strategy

אשר תממש את המטודה כך שתחזיר true אם המזג האוויר מתאים לרכב על אופניים ( בין 10-23 מעלות ) וfalse - אחרת.

**public** **class** SkiStrategy **implements** Strategy

אשר תממש את המטודה כך שתחזיר true אם המזג האוויר מתאים לסקי , תחזיר true אם יש פחות מ – 5 מעלות ו-false אחרת.

1. יש לכתוב את המחלקת ה- Context באופן הבא:
2. **public** **class** Context {
3. **int** temperature;
4. Strategy strategy;
5. **public** Context(**int** temperature, Strategy strategy) {
6. **this**.temperature = temperature;
7. **this**.strategy = strategy;
8. }
9. **public** **void** setStrategy(Strategy strategy) {
10. **this**.strategy = strategy;
11. }
12. **public** **int** getTemperature() {
13. **return** temperature;
14. }
15. **public** **boolean** getResult() {
16. **return** strategy.checkTemperature(temperature);
17. }
18. }

ולכתוב –main הבודק את שימושה.