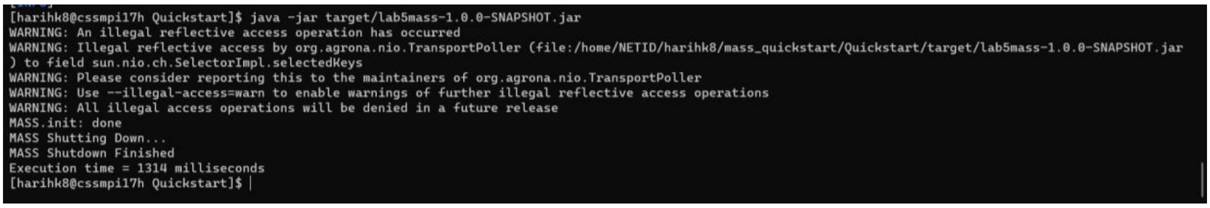
LAB 5 - Report : Hari Krishnan Raj Kumar

Ouput:



```
[harihk8@cssmpi17h Quickstart]$ java -jar target/lab5mass-1.0.0-SNAPSHOT.jar
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.agrona.nio.TransportPoller (file:/home/NETID/harihk8/mass_quickstart/Quickstart/target/lab5mass-1.0.0-SNAPSHOT.jar
) to field sun.nio.ch.SelectorImpl.selectedKeys
WARNING: Please consider reporting this to the maintainers of org.agrona.nio.TransportPoller
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
MASS.init: done
MASS Shutting Down...
MASS Shutdown Finished
Execution time = 1314 milliseconds
[harihk8@cssmpi17h Quickstart]$ |
```

Code:

```java
package edu.uwb.css534;
import edu.uw.bothell.css.dsl.MASS.*;
import edu.uw.bothell.css.dsl.MASS.logging.LogLevel;
import edu.uw.bothell.css.dsl.MASS.Agents;
import edu.uw.bothell.css.dsl.MASS.MASS;
import edu.uw.bothell.css.dsl.MASS.Places;
import java.util.*;
import java.io.*;
import java.lang.reflect.Array;
import java.util.Arrays;

public class lab5 {

    private static final String NODE_FILE = "nodes.xml";
    public static void main(String[] args) throws Exception {
        MASS.setNodeFilePath(NODE_FILE);
        MASS.setLoggingLevel(LogLevel.DEBUG);
        MASS.init();
        Places matrix = new Places( 1, Matrix.class.getName(), null, 10, 10);
//null as no arguments are passed in
        Agents worker = new Agents( 2, Worker.class.getName(), null, matrix,
2); //null as no arguments are passed in
        worker.callAll( Worker.goElsewhere_ );
        worker.manageAll( );
        MASS.finish();
    }


}
```

Matrix.java

```java
package edu.uwb.css534;

import edu.uw.bothell.css.dsl.MASS.*;
import java.util.*;

public class Matrix extends Place {

    public Matrix() { }
    public Matrix(Object obj) { }
    public Object callMethod(int method, Object o) {
        switch (method) {
            default:
                return new String("Unknown Method Number: " + method);

        }

    }
}
```

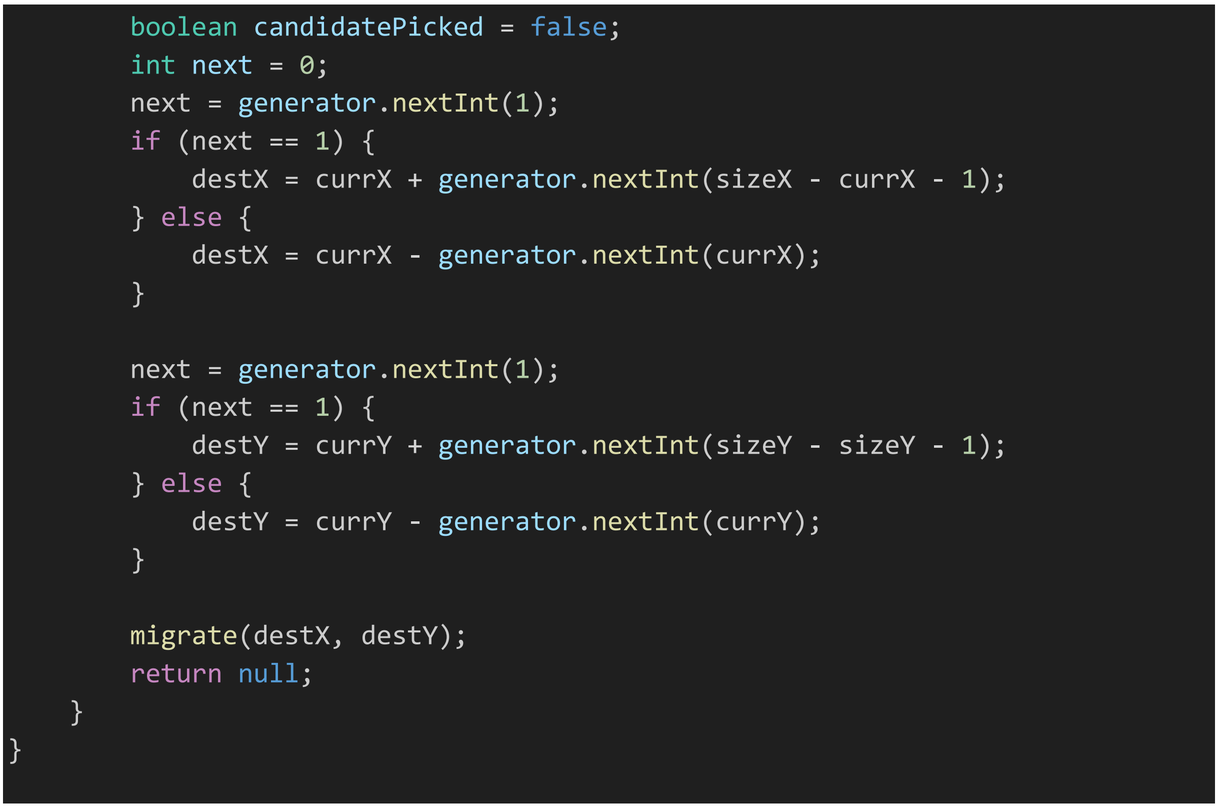
Worker.java

```java
package edu.uwb.css534;
import edu.uw.bothell.css.dsl.MASS.*;
import java.util.*;

public class Worker extends Agent {

    public static final int goElsewhere_ = 0;

    public Worker() {
        super();
    }
    public Worker(Object object) {
        super();
    }
    public Object callMethod(int funcId) {
        switch (funcId) {
            case goElsewhere_:
                return goElsewhere();
        }
        return null;
    }
    public Object goElsewhere() {
        int destX = 0;
        int destY = 0;
        int min = 1;
        int currX = getPlace().getIndex()[0], currY =
getPlace().getIndex()[1];
        int sizeX = getPlace().getSize()[0], sizeY = getPlace().getSize()[1];
        Random generator = new Random();
```



```
        boolean candidatePicked = false;
        int next = 0;
        next = generator.nextInt(1);
        if (next == 1) {
            destX = currX + generator.nextInt(sizeX - currX - 1);
        } else {
            destX = currX - generator.nextInt(currX);
        }

        next = generator.nextInt(1);
        if (next == 1) {
            destY = currY + generator.nextInt(sizeY - sizeY - 1);
        } else {
            destY = currY - generator.nextInt(currY);
        }

        migrate(destX, destY);
        return null;
    }
}
```