

Name – Rajdeep Jaiswal  
Semester – 3<sup>rd</sup>  
Branch – CSE Btech

Q. Create a class of SalesPersons as a thread that will display five salespersons names.

2. Create a class as Days as other Thread that has an array of seven days.

3. Call the instance of SalesPersons in Days and start both the threads

4. suspend SalesPersons on Sunday and resume on Wednesday

Note: use suspend, resume methods from the thread

CODE:

#### SalesPersonDaysThreadTask2.java

```
package com.company;

public class SalesPersonDaysThreadTask2 {
    public static void main(String[] args) {
        SalesPerson person = new SalesPerson();
        Thread t = new Thread(new Days(person));
        t.start();
    }
}
```

#### SP.java

```
package com.company;

import java.util.Random;

class SalesPerson implements Runnable {
    private String[] salesPersonName = { "sp1", "sp2", "sp3", "sp4", "sp5",
    "sp6", "sp7", "sp8" };
    private static boolean stopFlag = true;

    Thread th;

    public SalesPerson() {
        super();
        th = new Thread(this);
        th.start();
    }

    @Override
    public void run() {
```

```

        while (stopFlag) {
            try {
                int index = new Random().nextInt(7);
                System.out.println(salesPersonName[index]);
                Thread.sleep(500);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }

    public boolean isStopFlag() {
        return stopFlag;
    }

    public void setStopFlag(boolean stopFlag) {
        SalesPerson.stopFlag = stopFlag;
    }

    public void suspend() {
        setStopFlag(false);
    }

    public void resume() {
        setStopFlag(true);
    }
}

```

## Days.java

```

package com.company;

class Days implements Runnable {
    private String[] days = { "Sunday", "Monday", "Tuesday", "Wednesday",
    "Thursday", "Friday", "Saturday" };
    private static boolean stopFlag = false;
    SalesPerson salesPerson;

    public Days(SalesPerson salesPerson) {
        super();
        this.salesPerson = salesPerson;
        Thread t = new Thread(this);
        t.start();
        salesPerson.suspend();
    }

    @Override
    public void run() {
        synchronized (this) {
            try {
                while (!stopFlag) {
                    for (String day : days) {
                        System.out.println(day);
                        if (day.equals("Sunday") || day.equals("Monday") ||

```

```
day.equals("Tuesday")) {
    salesPerson.suspend();
} else if (day.equals("Wednesday")) {
    salesPerson.resume();
    Thread.sleep(2500);
} else {
    Thread.sleep(2500);
}
}
salesPerson.suspend();
stopFlag = true;
}
} catch (Exception e) {
    e.printStackTrace();
}
}
}
```

**OUTPUT:**

sp4  
Sunday  
Monday  
Tuesday  
Wednesday  
sp7  
sp6  
sp1  
sp7  
Thursday  
sp7  
sp6  
sp2  
sp1  
sp3  
Friday  
sp1  
sp6  
sp7  
sp3  
sp7  
Saturday  
sp7  
sp7