ID: 1930319

Name: Mohammad Arshad Hossain Ratul

CSE 213 Midterm

Sec-1

Project Topic: Textile Factory

1) a)

	User		Goal
1	M.D	————	check ~~g~~ progress report.
2	Factory manager	————	Production update daily.
3	Clients	————	Place orders.
4	Supplier	————	~~Set~~ check requirement
5	Government regulator	————	Check legal papers.

b) User: M.D:

Goal: Check progress report.

Workflow

- After the user login, the user will see the UI of the dashboard.
- The dashboard will have an option check progress report as a button. When the user clicks the button, it takes the user to another UI.

ID: 1930319

- This page will have three fields: Daily, Monthly and yearly and have clickable buttons with same name.
- Clicking on ~~Daily~~ Daily will show graphical reprentation of the work progress live, on another UI.
- Clicking on Monthly will show graphical represantation of ~~that~~ the work progress of the month on another UI.
- Clicking on yearly will show graphical representation of the work done on that perticular year.

~~• All the three fields Daily, Monthly and yearly will have~~

- Daily will have fields: ~~next, previous,~~ remarks, wools, silk, day, quantity etc.
- Monthly will have fields: Daily, remarks, month.
- yearly will have fields: Monthly, remarks, year, etc.
- Every UI will contain a back (button) which will bring back to the initial UI of check progress report.

```
(2) Class DailyResults
     {   Private:
           string remarks
           int quantity, wools, silk etc ;
           int* day;

         Public:
          void next (   ) ;
           {
             cout << show next day :
           }
          void previous ( )
           {
             cout << show previous day:
           }
          void show report ~~(int)~~ ( )
           {
             cout << show report ;
           }
     };    friend show reports( ) ;
    Class MonthlyResults
     {  Private:
          Daily Results* Daily; int month
          string remarks ;
        Public:
          void next ( )
          {  go to next month
          }
```

```
void previous( )
{ go to previous month;
}

void show reports( )
{
  cout<<show reports;
}
 Friend ~~show friends(~~ Show reports():
};

Class YearResults
{ Private:
  MonthlyResults* Monthly ;
  ~~Ba~~ int year;
  String remarks :

 Public:
  void next ( )
  { go to next ~~da~~ year;
  }
  void previous( )
  { go to previous year;
  }
```

1930319

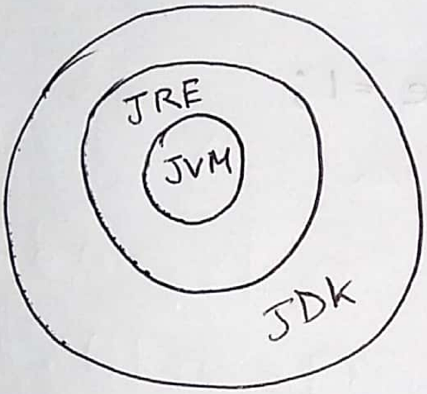
5

```
void show reports()
 { cout << show reports;
 }

};
```

2

a)



Sourse code

↓

JDK

↓

Machine code

Java converts every sourse code to machine code using JVM of that platform.

(b) That method package cannot be used by other classes.

```
Public class Ratul
{       class Food
          {  int apple =1;
          }
};

Public class GetFood
{       import Ratul;
Public Class FoodNo.
       {    Ratul. apple = 5;
       }
};
```

This cannot be exucuted as class food is not public.

(c) . ~~Yes, as "student asif" is a string~~

No, as student is a class and ~~a~~ Asif is the name of the class. In c++ it is global but in Java it can be obseect of another class and can be public ~~as w.~~ or pakage,

```
① Class Array
  { int* dataPtr;
    int noofelements ; float avg ;

    Public:
    void setdata( Array ofArray a, int n},int x)
     {     setdata ptr ( n ) ;
           for (int i=0 ; i<n ; i++)
            {    ~~a.dataPtr[i] = rand()100~~
                a.dataPtr[i] = rand() % x ;
            }
     }
   void show data(~~PtrToArray~~E int n )
   {   float a = 0;
        for (int i=0 ;i<n ;i++)
        {   ~~dataptr[i]~~.
            cout<< dataptr[i] << "," ;
          a=a+avgData(n);
        }
        cout << "avg:" << a/n ;
   }
   void set dataptr (int n)
   {  noofelements=n;
      dataptr = new int[n];
   }
};
```

```
Class Array of Arrays{
{      Array* PtrToArrays;
       int noofArrays;
                                           100
       ~~PtrToArrays = new Array[noofArrays]~~;

       Public:
       void setArrayOfArrays (int, n , int max)
        {       set ptr to Array(n    );
             for (int i=0 ; i<n;i++)
               {
                  Ptr ToArrays[i] . setdata(Ptr to Array[i], n, max);

               }
        }

      void showArrayOfArrays ( ~~int n~~ )
       {     Float ave = 0;
             for(int i=0; i< no of Arrays ;i++)
               {   Ptr To Array[i].show data(n);
                   ~~showdata(PtrToArray[i], n);~~

                   ave = ave + get avg();

               }
             cout << "avg: " << ave/n ;
       }
      void set ptr toArray (int n).
      {     no.of array = n ;

            PtrToArrays = new Array[n];
}; }}
```