Class Employee
This class supports enquiries entered by the user and returns the selection chosen the user.

//Instance variables
private boolean present;
private ArrayList<List<String>> dayPresent;
private ArrayList<List<Double>> hours;

//Constructor
public Employee(ArrayList<List<String>> dayPresent, ArrayList<List<Double>> hours)
//initialize variables and objects and create a new object. Initialise the Shifts employee object to reference the data of a particular employee then

//Methods
public boolean getPresentDay(String Date)
//check if employee was present on a given day (i.e. going through the list and searching for Date)
 and return true if found.

public String getDetailsOfShifts(String Date)
//return details of a shift worked on a given day.

public double TotalHours(int week)
//Calculate the total hours worked during a given week and return the sum of those hours

public int ShiftsWorked(int week)
// return the number of shifts worked during a given week.

Class Shifts
//this class is used to store the shifts date and hours in a list format. So that the employee class may produce results to the enquiries asked by user.

//Instance Variables
private ArrayList<List<Double>> hours;
private ArrayList<List<String>> date;

//Constructor
public Shifts( )
//invisible constructor

//Methods
public void UpdateShifts(String date, double hours)
/*Update the objects for an employee's shifts for a given week. Using list objects to update the dates and hours of shifts. Remember in your 2D list should be a list with lists weeks and same goes for hours (Hint: there cannot be repetition of a date in a list of dates of the same week and list of hours should be the same index as the date).*/

public ArrayList<List<Double>> getListOfHours()
//return the list of hours of the shifts of an employee.
public ArrayList<List<String>> getListOfDates()
//return the list of dates of the shifts of an employee.

public String DisplayShiftDetails(String date)
//return details of shift worked on a given day(i.e. hours, Sign in and Sign out time).

public double getTotalHoursWorked(int week)
//Calculate the total hours worked for the given week. Go through the list and sum up the hours for the given week and return.

public int getNumberShifts(int week)
//return the number of shifts in a given week.

Class Day

//Instance Variables
private int year;
private int month;
private String day;
private Shifts day;

//Constructor
public Day(int year, int month, String day)
//initialize the date of the shift and object

//Methods
public string toDate()
//return in date format.
public void totalHours(String Sign_in, String Sign_out)
//Calculate the number of hours worked for a day/a shift and send date and hours worked in a day to shift class to update.(Hint: convert string to double)

public class EmployeeDriver
//This class is a driver for the employee class acts as a driver for the program process.
//Method
public static void main(String[] args)
/* The driver provides the user with a platform to enter the enquiry they want to receive, and the company to record the shifts date and times*/
Using a while loop you can prompt the user for an employee's date, sign in and sign out time. Then sends to the Shifts class to update the shifts.

//objects to use to record shifts
Day NameEmployee = new Day(int year, int month, String day);
//call methods inside to record the sign in and sign out times.
Shifts NameEmployee = new Shifts();
/*Every new shift is recorded is added to the employee's shifts
Every other employee another object is created*/
//objects created for enquiry process for a particular employee.
Employee NameEmployee = new Employee(NameEmployee.
getListOfDates(),NameEmployee. getListOfHours());
//call methods inside to support the enquiries for an employee.