Assume you have the following declarations:

#include <iostream>

#include <string>

using namespace std;

//////////////////////////////////////////////////

class staff {

private :

 string name ;

public:

 staff(string n) { name=n; }

 void display(){cout <<"name = " <<name;}

 };

//=============================================

class SalaryEmployee : public staff {

private :

 int salary ;

public:

 SalaryEmployee (int s , string n) :staff(n)

 { salary = s;}

 // display ()

};

//=============================================

class HourlyEmployee : public staff{

 private :

 int hourly\_pay ;

public :

 HourlyEmployee (int h , string n) :staff(n){hourly\_pay=h;}

 // display ()

};

//=============================================

class ContractEmployee

{

};

//=============================================

//print function

///////////////////////////////////

void main (){

}

* **Complete the missing display () function in the classes**

**Add a fourth class:**

1. Create a class called **ContractEmployee** that inherits class Staff as public and has **contract duration** as a private element.
	1. The constructor for this class should initialize its new private elements from the arguments passed to it.
	2. The display function of this class should display its information

**In main:**

1. Create an array of 6 pointers to class Staff.
	1. Dynamically create objects of class **SalaryEmployee**, or **HourlyEmployee** or **ContractEmployee** .

Depending on user choice.

* 1. Fill the objects with information from user.
	2. Finally write a **function** (**regular function not in a class**) that prints a report of the 6 objects.