```

#ifndef MYHEADER_H_
#define MYHEADER_H_

//Preprocessor directives

#include<iostream> //for input and output
#include<iomanip>  //for output style
#include<string>   //for using string

//using the name space standard
using namespace std;


//class date: for defining the date
class date
{
//public parts containing the method functions of the class
public:

    //default constructor
    date();

    //destructor
    ~date();

//protected attributes of the class (accessible by derived classes)
protected:
    int month;  //PROCESS – for storing month
    int day;//PROCESS – for storing day
    int year;   //PROCESS – for storing year
};


//Class employee(derived from the class date):
//for setting and changing the attributes about the employees
class employee: protected date
{
//public parts containing the method functions of the class
public:
    //default constructor
    employee();


    //destructor
    ~employee();

    //method function for setting the name
    void setName(string empName);

    //method function for setting the Id
    void setId(string empId);

    //method function for setting the phone number
    void setPhoneNumber(string Number);

    //method function for setting the age
    void setAge(int empAge);

    //method function for setting the gender
    void setGender(char sex);

    //method function for setting the job title
    void setJobTitle(string title);

    //method function for setting the salary
    void setSalary(double income);

    //method function for setting the hire date
    void setDate(int startDay, int startMonth, int startYear);

    //method function for printing the attributes of the class
    void print();

//protected attributes of the class (accessible by derived classes)
protected:
    string  name;       //PROCESS - storing the name
    string  id;         //PROCESS - storing the id number
    string  phoneNum;   //PROCESS - storing the phone number
    int     age;    //PROCESS - storing the age
    chargender;     //PROCESS - storing the gender
    string  jobTitle;   //PROCESS - storing the job title
    double  salary;     //PROCESS - storing the salary
    datestartDate;  //PROCESS - storing the hire date
};

//class programmer(derived from the class employee):
//for setting and changing the attributes about the programmer
class programmer: protected employee
{
//public parts containing the method functions of the class
public:
    //default constructor
    programmer();


    //destructor
    ~programmer();

    //method for setting the department number
    void setDepartmentNum(string depNum);

    //method for setting the supervisor's name
    void setSupName(string supervisor);

    //method for setting the salary increase percentage
    void setPercentage(int percent);

    //method for setting if the person knows c++
    void cppIdentifier(bool cppIdent);
    //method for setting if the person knows java
    void javaIdentifier(bool javaIdent);

    //method for printing the attributes of the class
    void printProgrammer();

//private part of the class; containing the attributes of the class
private:
    string  depNumber;      //PROCESS - storing the department number
    string  supName;    //PROCESS - storing the supervisor's name
    float   percSalaryInc;  //PROCESS - storing the salary increase percentage
    boolcpp;        //PROCESS - storing if the person knows c++
    booljava;           //PROCESS - storing if the person knows java

};

//class softArch(derived from the class employee):
//for setting and changing the attributes about the software Architects
class softArch: protected employee
{
//public parts containing the method functions of the class
public:

    //default constructor
    softArch();

    //destructor
    ~softArch();

    //method for setting the department number
    void setDepartmentNum(string departmnet);

    //method for setting the supervisors name
    void setSupName(string supervisor);

    //method for setting the salary increase percentage
    void setPercentage(float percent);

    //method for setting the experience
    void setExperience(int empYears);

    //method for printing the attributes of the class
    void printSoftArch();

//private part of the class; containing the attributes of the class
private:
    string  depNumber;      //PROCESS - storing the department number
    string  supName;    //PROCESS - storing the supervisor's name
    float   percSalaryInc;  //PROCESS - storing the salary increase percentage
    int     experience;     //PROCESS - storing the experience
};
```

```c
#endif /* MYHEADER_H_ */



```