

MyHeader.h

```
1
2 #ifndef MYHEADER_H_
3 #define MYHEADER_H_
4
5 //Preprocessor directives
6
7 #include<iostream> //for input and output
8 #include<iomanip>   //for output style
9 #include<string>    //for using string
10
11 //using the name space standard
12 using namespace std;
13
14
15 //class date: for defining the date
16 class date
17 {
18 //public parts containing the method functions of the class
19 public:
20
21     //default constructor
22     date();
23
24     //destructor
25     ~date();
26
27 //protected attributes of the class (accessible by derived classes)
28 protected:
29     int month; //PROCESS - for storing month
30     int day;   //PROCESS - for storing day
31     int year;  //PROCESS - for storing year
32 };
33
34
35 //Class employee(derived from the class date):
36 //for setting and changing the attributes about the employees
37 class employee: protected date
38 {
39 //public parts containing the method functions of the class
40 public:
41     //default constructor
42     employee();
43
44
45     //destructor
46     ~employee();
47
48     //method function for setting the name
49     void setName(string empName);
50
51     //method function for setting the Id
52     void setId(string empId);
53
54     //method function for setting the phone number
55     void setPhoneNumber(string Number);
```

MyHeader.h

```
56
57 //method function for setting the age
58 void setAge(int empAge);
59
60 //method function for setting the gender
61 void setGender(char sex);
62
63 //method function for setting the job title
64 void setJobTitle(string title);
65
66 //method function for setting the salary
67 void setSalary(double income);
68
69 //method function for setting the hire date
70 void setDate(int startDay, int startMonth, int startYear);
71
72 //method function for printing the attributes of the class
73 void print();
74
75 //protected attributes of the class (accessible by derived classes)
76 protected:
77     string name;           //PROCESS - storing the name
78     string id;             //PROCESS - storing the id number
79     string phoneNum;       //PROCESS - storing the phone number
80     int age;               //PROCESS - storing the age
81     char gender;           //PROCESS - storing the gender
82     string jobTitle;       //PROCESS - storing the job title
83     double salary;         //PROCESS - storing the salary
84     date startDate;        //PROCESS - storing the hire date
85 };
86
87 //class programmer(derived from the class employee):
88 //for setting and changing the attributes about the programmer
89 class programmer: protected employee
90 {
91 //public parts containing the method functions of the class
92 public:
93     //default constructor
94     programmer();
95
96
97     //destructor
98     ~programmer();
99
100 //method for setting the department number
101 void setDepartmentNum(string depNum);
102
103 //method for setting the supervisor's name
104 void setSupName(string supervisor);
105
106 //method for setting the salary increase percentage
107 void setPercentage(int percent);
108
109 //method for setting if the person knows c++
110 void cppIdentifier(bool cppIdent);
```

MyHeader.h

```
111
112 //method for setting if the person knows java
113 void javaIdentifier(bool javaIdent);
114
115 //method for printing the attributes of the class
116 void printProgrammer();
117
118 //private part of the class; containing the attributes of the class
119 private:
120     string  depNumber;        //PROCESS - storing the department number
121     string  supName;         //PROCESS - storing the supervisor's name
122     float   percSalaryInc;    //PROCESS - storing the salary increase percentage
123     boolcpp;                 //PROCESS - storing if the person knows c++
124     booljava;                //PROCESS - storing if the person knows java
125
126 };
127
128 //class softArch(derived from the class employee):
129 //for setting and changing the attributes about the software Architects
130 class softArch: protected employee
131 {
132 //public parts containing the method functions of the class
133 public:
134
135     //default constructor
136     softArch();
137
138     //destructor
139     ~softArch();
140
141     //method for setting the department number
142     void setDepartmentNum(string departmnet);
143
144     //method for setting the supervisors name
145     void setSupName(string supervisor);
146
147     //method for setting the salary increase percentage
148     void setPercentage(float percent);
149
150     //method for setting the experience
151     void setExperience(int empYears);
152
153     //method for printing the attributes of the class
154     void printSoftArch();
155
156 //private part of the class; containing the attributes of the class
157 private:
158     string  depNumber;        //PROCESS - storing the department number
159     string  supName;         //PROCESS - storing the supervisor's name
160     float   percSalaryInc;    //PROCESS - storing the salary increase percentage
161     int     experience;       //PROCESS - storing the experience
162 };
163
164
165
```

MyHeader.h

```
166 #endif /* MYHEADER_H_ */  
167  
168  
169
```