// https://github.com/mannanarama/CS-256

#include "stdafx.h"

#include <iostream>

#include <string>

using namespace std;

class Employee

{

string name;

int number;

string date;

public:

Employee(string n, int num, string d)

{

name = n;

number = num;

date = d;

}

Employee()

{

name = " ";

number = 0;

date = " ";

}

class InvalidNumber

{

};

void setName(string n)

{

name = n;

}

void setNumber(int num)

{

if (number > 0 && number < 9999)

{

number = num;

}

else

{

throw InvalidNumber();

}

}

void setDate(string d)

{

date = d;

}

string getName()

{

return name;

}

int getNumber()

{

return number;

}

string getDate()

{

return date;

}

};

class ProductionWorker : public Employee

{

int shift;

double payRate;

public:

ProductionWorker(string aName, int aNumber, string aDate, int aShift, double aPayRate)

{

shift = aShift;

payRate = aPayRate;

}

ProductionWorker() : Employee()

{

shift = 0;

payRate = 0.0;

}

class InvalidShift

{

};

class InvalidPayRate

{

};

void setShift(int s)

{

if (shift == 1 || shift == 2)

{

shift = s;

}

else

{

throw InvalidShift();

}

}

void setPayRate(double pr)

{

if (payRate > 0)

{

payRate = pr;

}

else

{

throw InvalidPayRate();

}

}

int getShift()

{

return shift;

}

string getShiftName()

{

if (shift == 1)

{

return "DAY";

}

else if (shift == 2)

{

return "NIGHT";

}

}

double getPayRate()

{

return payRate;

}

};

int main()

{

try

{

Employee emp;

emp.setNumber(-1234);

}

catch (Employee::InvalidNumber)

{

cout << "Error: Negative value imput\n";

}

ProductionWorker pw;

try

{

pw.setShift(3);

}

catch(ProductionWorker::InvalidShift)

{

cout << "Error: Invalid Shift number\n";

}

try

{

pw.setPayRate(-10.50);

}

catch (ProductionWorker::InvalidPayRate)

{

cout << "Error: Invalid Pay Rate\n";

}

}