Put at least 5 doxygen commands within the inherit2 example. Generate the corresponding documentation.

Work with a partner.

Illustrate the results.

//inherit2.h header file for the PersonType class

**#ifndef** H\_PersonType

**#define** H\_PersonType

**#include** <string>

**using** **namespace** std;

**class** personType

{

**public**:

**void** **print**() **const**;

//Function to output the first name and last name

//in the form firstName lastName

**void** **setName**(string first, string last);

//Function to set firstName and lastName according to

//the parameters

//Post: firstName = first; lastName = last;

**void** **getName**(string& first, string& last);

//Function to return firstName and lastName via the parameters

//Post: first = firstName; last = lastName;

**personType**(string first, string last);

//Constructor with parameters

//Set firstName and lastName according to the parameters

//Post: firstName = first; lastName = last;

**personType**();

//Default constructor;

//Intialize firstName and lastName to empty string

//Post: firstName = ""; lastName = "";

**private**:

string firstName; //store the first name

string lastName; //store the last name

};

**#endif**

//inherit2i.cpp implementation file for the PersonType class

**#include** <iostream>

**using** **namespace** std;

**#include** <string>

**#include** "inherit2.h"

**using** **namespace** std;

**void** **personType::print**() **const**

{

cout<<firstName<<" "<<lastName;

}

**void** **personType::setName**(string first, string last)

{

firstName = first;

lastName = last;

}

**void** **personType::getName**(string& first, string& last)

{

first = firstName;

last = lastName;

}

//constructor with parameters

**personType::personType**(string first, string last)

{

firstName = first;

lastName = last;

}

**personType::personType**() //default constructor

{

firstName = "";

lastName = "";

}

// header file for the partTimeEmployee

**#include** "inherit2.h"

**class** partTimeEmployee: **public** personType

{

**public**:

**void** **print**();

//Function to output the first name, last name, and

//the wages in the form:

//firstName lastName wages are $$$$.$$

**double** **calculatePay**();

//Function to calculate and return the wages

**void** **setNameRateHours**(string first, string last,

**double** rate, **double** hours);

//Function to set the first name, last name, payRate,

//and hoursWorked according to the parameters.

//The parameters first and last are passed to the

//base class. payRate = pay; hoursWorked = hours;

**partTimeEmployee**(string first, string last,

**double** rate, **double** hours);

//Constructor with parameters

//Set the first name, last name, payRate, and

//hoursWorked according to the parameters.

//Parameters first and last are passed to the

//base class. payRate = pay; hoursWorked = hours;

**partTimeEmployee**();

//Default constructor

//Set the first name, last name, payRate, and

//hoursWorked to the default values.

//The first name and last name are initialized to an empty

//string by the default constructor of the base class.

//payRate = 0; hoursWorked = 0;

**private**:

**double** payRate; //store the pay rate

**double** hoursWorked; //store the hours worked

};

//Implementation File partTimeEmployee class

**#include** <iostream>

**#include** "inherit2.h"

**#include** "partTimeEmployee.h"

**using** **namespace** std;

**void** **partTimeEmployee::print**()

{

personType::print(); //print the name of the employee

cout<<" wages are : "<<calculatePay()<<**endl**;

}

**double** **partTimeEmployee::calculatePay**()

{

**return** (payRate \* hoursWorked);

}

**void** **partTimeEmployee::setNameRateHours**(string first,

string last, **double** rate, **double** hours)

{

personType::setName(first,last);

payRate = rate;

hoursWorked = hours;

}

**partTimeEmployee::partTimeEmployee**(string first, string last,

**double** rate, **double** hours)

: personType(first, last) //constructor with parameters

{

payRate = rate;

hoursWorked = hours;

}

**partTimeEmployee:: partTimeEmployee**() // default constructor

{

payRate = 0;

hoursWorked = 0;

}

//client for TimeEmployee

**#include** <iostream>

**#include** "inherit2.h"

**#include** "partTimeEmployee.h"

**using** **namespace** std;

**int** **main**()

{

personType newPerson;

partTimeEmployee newEmployee("John","Smith",7.50,56);

partTimeEmployee employee;

newEmployee.print();

employee.setNameRateHours("Rachel", "Moore",9.75, 45);

employee.print();

**return** 0;

}

output

John Smith wages are : 420

Rachel Moore wages are : 438.75