**TypPiece.h :**

#pragma once

#include <string>

#include <iostream>

using namespace std;

class TypPiece

{

protected:

string nom;

int code;

public:

TypPiece(string str = "", int c = 0);

string getNom() { return this->nom; }

int getCode() { return this->code; }

**//============Question 3==============**

void affiche();

};

**TypPieceProd.h :**

#pragma once

#include "TypPiece.h"

class TypPieceProd : public TypPiece

{

protected:

int nbBruts, nbUsi, nbRebus; // Nombre de piéces brutes, usinées et le rebus

**//==========Question 2============**

static int reference;

public:

TypPieceProd(string leNom = "PROD", int leCode = -1, int leNbBruts = 0, int leNbUsi = 0, int leNbRebus = 0);

~TypPieceProd();

int getNbRebus() { return this->nbRebus; } // Accesseur au nombre de rebus

**//==========Question 3============**

void affiche();

**//==========Question 4============**

void afficheRef();

**//==========Question 5============**

void affichage();

};

**TypPiece.cpp :**

#include "TypPiece.h"

**//==========Question 1============**

TypPiece::TypPiece(string str, int c)

{

this->nom = str;

this->code = c;

}

**//==========Question 3============**

void TypPiece::affiche()

{

cout << "Le nom est " << this->nom << endl << "Le code est " << this->code << endl;

}

**TypPieceProd.cpp :**

#include "TypPieceProd.h"

**//==========Question 2============**

int TypPieceProd::reference = 0;

TypPieceProd::TypPieceProd(string leNom, int leCode, int leNbBruts, int leNbUsi, int leNbRebus) : TypPiece(leNom, leCode)

{

this->nbBruts = leNbBruts;

this->nbUsi = leNbUsi;

this->nbRebus = leNbRebus;

this->reference++;

}

TypPieceProd::~TypPieceProd()

{

this->reference--;

}

**//==========Question 3============**

void TypPieceProd::affiche()

{

TypPiece::affiche();

cout << "le nombre de rebus est de : " << nbRebus << endl;

}

**//==========Question 4============**

void TypPieceProd::afficheRef()

{

cout << "Le reference est de : " << reference << endl;

}

**//==========Question 5============**

void TypPieceProd::affichage()

{

TypPiece::affiche();

cout << "Le nombre de pieces bruts est de : " << nbBruts << endl;

cout << "Le nombre de piece usinee est de : " << nbUsi << endl;

cout << "le nombre de rebus est de : " << nbRebus << endl;

}

**Main.cpp :**

#include <iostream>

#include <conio.h>

#include <string>

#include "TypPieceProd.h"

using namespace std;

int main()

{

**//==========Question 3=============**

cout << "==========QUESTION 3==========\n" << endl;

TypPieceProd\* piece1;

piece1 = new TypPieceProd("T1", 1, 100, 50, 3);

piece1->affiche();

**//==========Question 4=============**

cout << endl << "==========QUESTION 4==========\n" << endl;

TypPieceProd\* piece2;

piece1->afficheRef();

piece2 = new TypPieceProd("T2", 2, 200, 150, 7);

piece2->affiche();

piece2->afficheRef();

delete piece2;

piece2->afficheRef();

**//==========Question 5=============**

cout << endl << "==========QUESTION 5==========\n" << endl;

int i, y = 1,val=1;

TypPieceProd\* piece[30];

for (i = 0; i < 30; i++)

{

string strDebut = "REF";

string strFin = to\_string(val);

piece[i] = new TypPieceProd(strDebut += strFin, y, 50, 0, 0);

y = y + 1;

val = val + 1;

}

for (i = 0; i < 30; i++)

{

piece[i]->affichage();

}

piece[30]->afficheRef();

for (i = 0; i < 30; i++)

{

delete piece[i];

}

piece[30]->afficheRef();

}