ASSIGNMENT NO: 3

Name : Krushna Maruti Diwate

Roll no : 16

Batch : S4

Date :

Imagine a publishing company which does marketing for book and audio cassette versions. Create a class publication that stores the title (a string) and price (type float) of publications. From this class derive two classes: book which adds a page count (type int) and tape which adds a playing time in minutes (type float). Write a program that instantiates the book and tape class, allows user to enter data and displays the data members. If an exception is caught, replace all the data member values with zero values.

Code:

```
#include <iostream>
#include <string>
using namespace std;

class pub
{
private:
string title;
float prices;

public:
pub()
 {
```

```cpp
title = "";
prices = 0.0;

}

void get_data()
{
cout<<"\nEnter title : ";
cin.ignore(); //clear input buffer
getline(cin,title);
cout<<"\nEnter Price : ";
cin>>prices;
}
void put_data()
{
cout<<"\n";
cout<<"\n Information : " << endl;
cout<<"\n Title : "<<title;
cout<<"\n Price : "<<prices;

}
};

class book: public pub
{
private:
int pages;
public:
book()
{
```

```cpp
pages = 0;
}

void get_data()
{
pub::get_data();
cout<<endl;
cout<<"Enter Page Count : \n";
cin>>pages;

}

void put_data()
{
pub::put_data();
try{
if(pages<0)
throw pages;}
catch(int f)
{
cout<<"\n error: pages not valid. "<<f;
pages=0;
}
cout<<"\n Pages Are : "<<pages;
}
};
class tape: public pub
{
private:
float playtime;
```

```cpp
public:
tape()
{
playtime=0.0;
}
void get_data()
{
pub::get_data();
cout<<"Enter Play Time Of Cassette \n";
cin>>playtime;
}
void put_data()
{
pub::put_data();
try
{
if(playtime<0.0)
throw playtime;
}
catch(float r)
{
cout<<"\n Error: Invalid Playtime :"<<playtime;
playtime=0.0;
}
cout<<"\n Playtime is : "<<playtime;
}
};

int main() //main func
{
```

```cpp
book b[10]; //array of object
tape t[10];
int choice=0,bookcount=0,tapeCount=0;
cout<<"---------------------------------";
do
{
cout<<"\n 1. Add Book ";
cout<<"\n 2. Display tape : ";
cout<<"\n 3. Display Book ";
cout<<"\n 4. Display tape ";
cout<<"\n 5. Exit : "<<endl;
cout<<"\n Enter Choice : ";
cin>>choice;
switch(choice)
{
case 1:
{
cout<<"\n------------------\n";
cout<<"Add Book : \n";
b[bookcount].get_data();
bookcount++;
break;
}
case 2:
{
cout<<"\n------------------\n";
cout<<"Add Tape : \n";
t[tapeCount].get_data();
tapeCount++;
break;
```

```cpp
}
case 3:
{
cout<<"\n (books)";
for(int j=0;j<bookcount;j++)
{
b[j].put_data();
}
break;
}
case 4:
{
cout<<"\n (tape)";
for(int j=0;j<tapeCount;j++)
{
t[j].put_data();
}
break;
}
case 5:
{
cout<<"**************** Program Exited Successfully ********************"<<endl;
exit(0);
}
default:
{
cout<<"\n Invalid";
}
}
}
```

```
while(choice!=5);
return 0;
}
```

Output:

```
--------------------------------
 1. Add Book
 2. Display tape :
 3. Display Book
 4. Display tape
 5. Exit :

 Enter Choice : 1
 -------------------
Add Book :

Enter title : Harry Potter
Enter Price : 499
Enter Page Count :
1099
1. Add Book
 2. Display tape :
 3. Display Book
 4. Display tape
 5. Exit :

 Enter Choice : 3
 (books)
```

Information :

Title : Harry Potter

Price : 499

Pages Are : 1099

1. Add Book

2. Display tape :

3. Display Book

4. Display tape

5. Exit :

Enter Choice : 5

***************** Program Exited Successfully *********************