

Practical Examination.

Roll No : 21143

Date : 21/12/2021

Name : Tanmay Karmarkar

Problem Statement :

Imagine a publishing company which does marketing for book & audio cassette versions. Create a class publication that stores the title (string) & price (float) of publications. From this class derive 2 classes : book which adds a page count (int) & tape which adds a play time in minutes (float).

Write a program that instantiates the book & tape class, allow user to enter data & displays the data members. If an exception is caught, replace all data members values with zero value. Do exception handling for all inputs.

Theory:

Class : Class is an user defined data type which can have data members. One can define functions in classes.

The data members can be accessed by creating instances of classes.

A class has 3 access specifiers:

1. Private : Members cannot be accessed (or viewed) from outside the class.
2. Protected : Members cannot be accessed from outside the class however can be accessed by inherited classes.
3. Public : Members accessible from outside of the class.

Base class: The class which is a subclass is called as derived class or base class.

Super class: The class from which base class is derived is a superclass.

Question Reciting:

Arrays: An array is a collection of items stored one after the other.

Pseudocode / Algorithm:

1. Define Publication class with title, price, title array, price array.
2. From that derive 2 class Book & Tape.
3. Declare function for adding record, updating record, deleting record, search record.
4. Declare page count, page count array.

A. void Add Record ()

1. Take input from user for title
2. Check whether only numbers are not entered
or if entered set all parameters to 0.
3. Take input of price and check for if price > 0 & character is not entered. If entered set all parameters to zero.
4. Take input for pagecount & do the same E&H for it.

B) void Display ()

1. Traverse from start till end of ~~poly~~ array.
2. If the value inside title array is not a NULL character print value of title array, price & pagecount array.
3. Increment counter by 1.
4. If counter = 0 display that no ~~boo~~ record present.

C) void ~~update~~ search (String a)

1. Traverse through title array & find if the string is present in the array.
2. If found print title [i], price [i], page count [i].

D) void update (String a)

1. Traverse through title array & find string in array.
2. If present ask user to input new name, price, page count.
3. Update it in the respective arrays.
4. If not found display name not found.

E) void delete (String a)

1. Traverse through title array & find string in array
2. If present set all the parameters to 0.
3. If not found display ~~boo~~ record not found.

Similar for functions present in Page class.

Test Cases.

| Description | Expected O/P | Actual O/P | Result. |
|--|--------------------------------|--|---------|
| 1. Create Book Name: 123 | Parameters set to 0 | Parameters set to 0 | Pass |
| 2. Create Book Name: rama Price: -2 | Parameters set to 0 | Parameters set to 0 | Pass |
| 3. Display Book No Book present | Database empty. | Database empty. | Pass |
| 4. Display Book | 1. Ramayan 2. 200 3. 300 | 1. Ramayan 2. 200 3. 300 | Pass |
| 5. Update Book Name: mahabharat | Record Not Present | Book Not present | Pass |
| 6. Update book Name: maha ramayan new: name: maha maha price: 204 pg count: 500 | - | - | Pass. |
| 7. Display Book. | 1. maha 2. 204 3. 500 | 1. maha 2. 204 3. 500 | Pass. |

| | | | | |
|----|---|----------------------------------|---|--------|
| 8. | Search Book Name: rama rama | Book not present | Book not present | Pass |
| 9 | Search Book Name : rama rama | 1. rama maha 2. 204 3. 500 | 1. rama maha 2. 204 3. 500 | Pass , |
| 10 | Exit | Thank You | Thank You | Pass. |

Conclusion:

We learnt at the implementation of
we learned about exception handling and concept
of super class, derived class & access specifics.