

Exercises

1. Define a class that holds information about a mobile phone device: model, manufacturer, price, owner, battery characteristics (model, hours idle and hours talk) and display characteristics (size and number of colors). Define 3 separate classes (class GSM holding instances of the classes Battery and Display).
2. Define several constructors for the defined classes that take different sets of arguments (the full information for the class or part of it). Assume that model and manufacturer are mandatory (the others are optional). All unknown data fill with null.
3. Add an enumeration BatteryType (Li-Ion, NiMH, NiCd, ...) and use it as a new field for the batteries.

Exercises (2)

4. Add a method in the `GSM` class for displaying all information about it. Try to override `ToString()`.
5. Use properties to encapsulate the data fields inside the `GSM`, `Battery` and `Display` classes. Ensure all fields hold correct data at any given time.
6. Add a static field and a property `iPhone4S` in the `GSM` class to hold the information about iPhone 4S.
7. Write a class `GSMTTest` to test the `GSM` class:
 - ♦ Create an array of few instances of the `GSM` class.
 - ♦ Display the information about the `GSMs` in the array.
 - ♦ Display the information about the static property `iPhone4S`.

Exercises (3)

8. Create a class `Call` to hold a call performed through a GSM. It should contain date, time, dialed phone number and duration (in seconds).
9. Add a property `CallHistory` in the `GSM` class to hold a list of the performed calls. Try to use the system class `List<Call>`.
10. Add methods in the `GSM` class for adding and deleting calls from the calls history. Add a method to clear the call history.
11. Add a method that calculates the total price of the calls in the call history. Assume the price per minute is fixed and is provided as a parameter.

Exercises (4)

12. Write a class `GSMCallHistoryTest` to test the call history functionality of the `GSM` class.
- ♦ Create an instance of the `GSM` class.
 - ♦ Add few calls.
 - ♦ Display the information about the calls.
 - ♦ Assuming that the price per minute is 0.37 calculate and print the total price of the calls in the history.
 - ♦ Remove the longest call from the history and calculate the total price again.
 - ♦ Finally clear the call history and print it.