# COMP8820  A2 - Timed Assessment

## Introduction

This is a timed assessment; you are expected to complete the tasks within 2 hours, but you can take longer if you need more time. You should submit your work before the deadline.

You will work with a given project. It is a simple simulation of robots that can chat with others. It consists of an abstract class named `Robot`, an interface named `Chatty`, and a concrete class named `ChatBot` that is incomplete.

## Important – Please read carefully

It is important that you should follow the instructions below:

- You should download the given project file `COMP8820-A2.zip` from Moodle and unzip the file before working on the tasks for A2 using BlueJ.
- You should ***not*** make any changes in both `Robot` and `Chatty.`
- In the `ChatBot` class you should ***not*** change the followings:
  - the declarations of the private fields: `level`, `friends`, `chatRecords`
  - the signature (i.e. header) of the class constructor
  - the method signature of `addFriend`
  - the method definitions of these methods:
    `getLevel`, `getFriends`, `getChatRecords`, `addChatRecord`, `equals`
- You should make sure that your program compiles without any syntax errors.

  ***Note****: marks will be deducted if your code doesn't compile with the given* `Robot` *and* `Chatty`*.*
- You should test your implementation thoroughly to make sure they work as expected.
- You should submit your work before the deadline. *No extension is allowed for the test.*

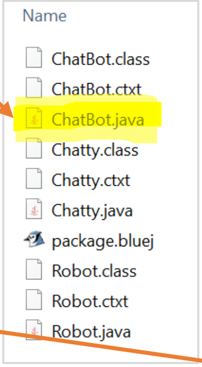
## Submission

The submission deadline is at 23:55 on Thursday 7 Dec 2023.

You should upload a *single* Java file, ChatBot.java, by clicking on "A2 Submission link" on the module Moodle page, before the deadline. *Please do not change the filename.*

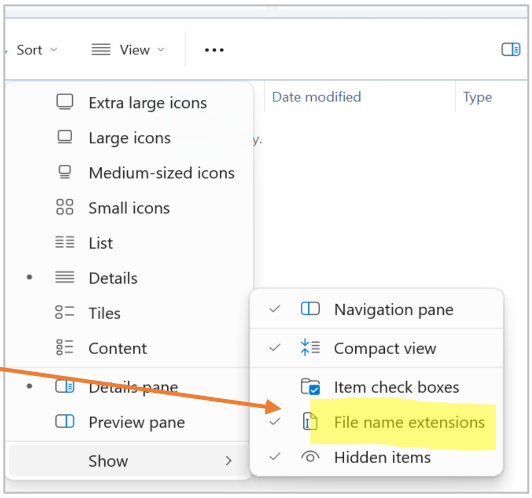
Note:

- You can find the file ChatBot.java inside your COMP8820-A2 project folder.
- If the file type `.java` is not visible in your file explorer window you should change the `View` setting by selecting "Show" and then the "File name extensions".



Name

ChatBot.class
ChatBot.ctxt
ChatBot.java
Chatty.class
Chatty.ctxt
Chatty.java
package.bluej
Robot.class
Robot.ctxt
Robot.java



## Tasks

You are expected to complete the implementation of the ChatBot class. In total there are 7 tasks as specified below.

1. [2 marks] The ChatBot class is a concrete subclass; it should inherit both Robot and Chatty.

   *Note:*

   - *the field level holds the AI level of the chatbot;*
   - *the field friends stores a list of the objects of ChatBot that are the chatbot's friends;*
   - *the field chatRecords records the chats (i.e. questions or answers) made by the chatbot.*

2. [4 marks] It has a single constructor that takes two parameters: a name and a number. Both fields friends and chatRecords are instantiated with an empty list.

   Replace //TODO with your own Java code to meet further requirements below:

   a) The name of the chatbot should be set to the given name.

   b) The level of the chatbot should be set to the given number if it is within the range [LEVEL_MIN, LEVEL_MAX]. If the given number is less than LEVEL_MIN, the level is set to LEVEL_MIN; if the number is greater than LEVEL_MAX, the level is set to LEVEL_MAX.

   *Note: LEVEL_MIN and LEVEL_MAX are defined in Chatty.*

3. [7 marks] The addFriend method should add the given object of ChatBot to the list friends. The list should not have any null values or duplicates.

   Complete the implementation of the addFriend method by replacing //TODO with your own Java code.

4. [3 marks] The implementation of the method hasAI of Robot should return true if the level of the chatbot is higher than LEVEL_MIN, otherwise it should return false.

5. [10 marks] The implementation of the method question of Chatty should return "Good" if the chatbot has no AI, otherwise it should return a question randomly selected from the collection QA. If the chatbot's level is LEVEL_MAX it should not ask the same question as the last one (i.e. the question before this one). The question should be recorded using the method addChatRecord.

   *Note: QA is the question-and-answer repertoires defined in Chatty.*

6. [11 marks] The implementation of the method answer of Chatty should return the answer to the given question from the collection QA if the chatbot has AI, otherwise it should return "Excellent" regardless of the given question. If the chatbot has AI but the given question is not in QA it should return "Interesting question" instead. The answer should be recorded using the method addChatRecord.

   *Note: QA is the question-and-answer repertoires defined in Chatty.*

7. [5 marks] Define a method named getChatStats that takes no parameter. It returns an array containing 2 numbers of int type: the first one is the total number of unique questions by the chatbot, and the second one is the total number of unique answers by the chatbot.