

The methodology of School 21 makes sense only if peer-to-peer reviews are done seriously. Please read all guidelines carefully before starting the review.

- Please, stay courteous, polite, respectful and constructive in all communications during this review.
- Highlight possible malfunctions of the work done by the person and take the time to discuss and debate it.
- Keep in mind that sometimes there can be differences in interpretation of the tasks and the scope of features. Please, stay open-minded to the vision of the other.
- If you have not finished the project yet, it is compulsory to read the entire instruction before starting the review.

## Guidelines

- Evaluate only the files that are in src folder on the GIT repository of the student or group.
- Ensure to start reviewing a group project only when the team is present in full.
- Use special flags in the checklist to report, for example, an “empty work” if repository does not contain the work of the student (or group) in the src folder of the develop branch, or “cheat” in case of cheating or if the student (or group) are unable to explain their work at any time during review as well as if one of the points below is not met. However, except for cheating cases, you are encouraged to continue reviewing the project to identify the problems that caused the situation in order to avoid them at the next review.
- Doublecheck that the GIT repository is the one corresponding to the student or the group.
- Meticulously check that nothing malicious has been used to mislead you.
- In controversial cases, remember that the checklist determines only the general order of the check. The final decision on project evaluation remains with the reviewer.

## MAIN PART



### Exercise 00 - Retirement Plan

Check that "main.py" is present

Check that the database of questions is present as a separate file in submission

Check that the questions are picked in random order every time you run the test

Check that the test ends after 10 question and gives you the binary result

Check that after each question you also need to type in values for respiration, heart rate, blushing and pupil size

Check that the final result actually depends on how you answer the questions and which metrics you submit

No



Yes

### Exercise 01 - Human Life

Check that the tests are included in the submission in `tests` directory

Check that tests cover both positive (proper behavior) and negative (handling bad input) cases

Check that current version of the code handles properly (without crashing) entering non-existent response number

Check that current version of the code handles entering invalid (text/char) values instead of numbers for metrics

Check that test explicitly checks positive scenario for human

Check that test explicitly checks positive scenario for replicant

### Exercise 02 - For the Future

Check that 'docs' directory is present

Check that documentation can be generated by running 'make html' inside 'docs' directory

Check that the title and logo for the project are clearly present in a generated documentation

Check that generated documentation includes a quickstart section explaining how to interact with the test

Check that generated documentation includes a reference section which is based on comments in the implementation

## Feedback



### Fails

### Comment