

Programming Game

Admin Manual

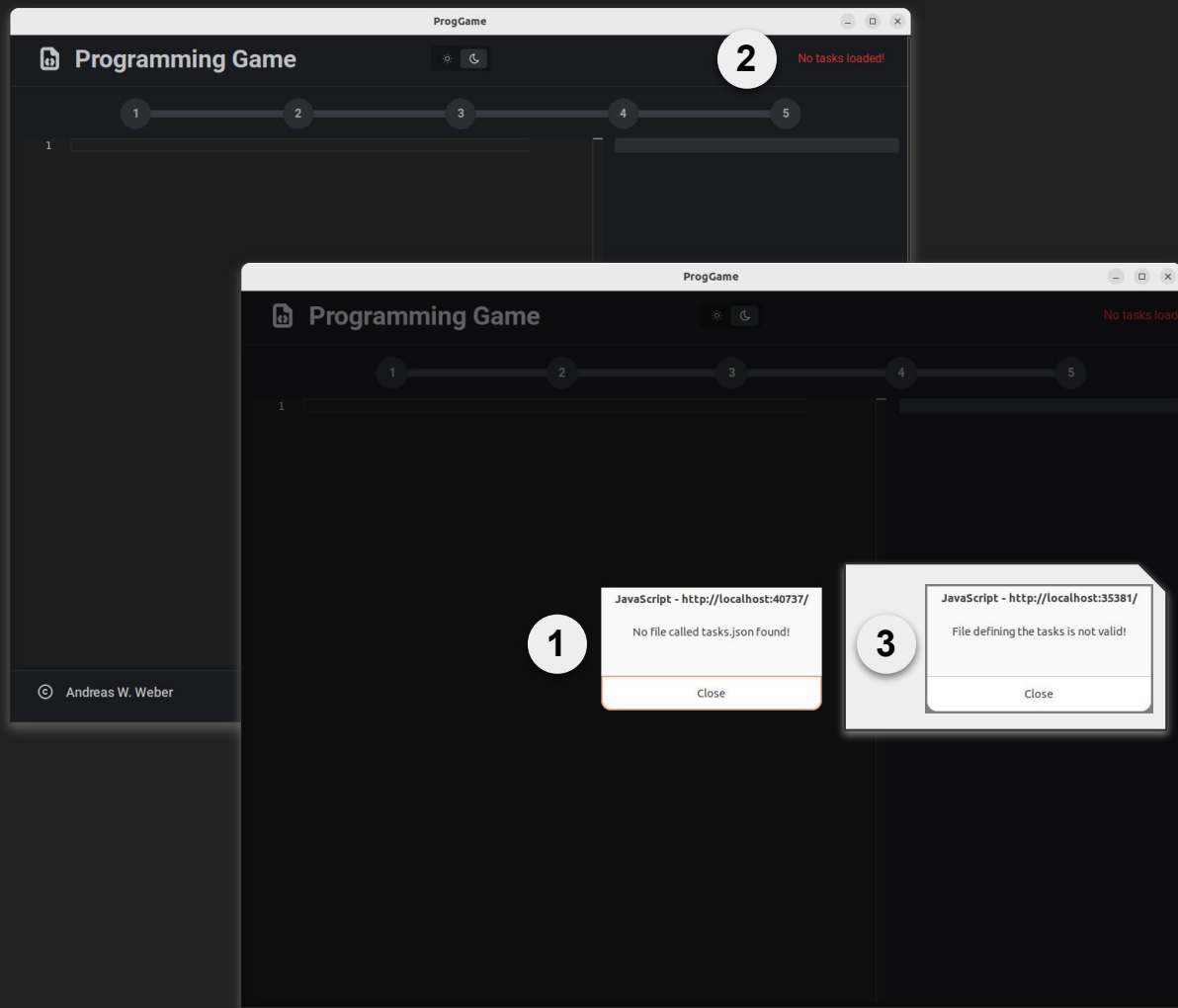
Prerequisites

(only tested on Ubuntu 22.04)

- python 3 is installed.
- minimal screen resolution: 1280 x 1024px
- use **chmod +x filename** before launching
- read the user manual
- launch **ProgGame-linux_x64** on a Linux Desktop Computer
- launch **ProgGame-linux_arm64** on a Raspberry Pi

Loading the Tasks

The tasks are loaded from the **tasks.json** file, which must be located in the folder where the program is started. If the program cannot find the file, an error message is displayed ❶ and the control elements for the challenge are hidden ❷. An invalid configuration of tasks.json also leads to an error message ❸. Note the following: ...



Loading the Tasks

... The file must represent a JSON array containing objects ④. Each object must contain the attributes **id**, **name**, **desc**, **code** and **test** ⑤. The data types must be correct. All attributes have the data type **string**, except **id**, which has the data type **number** ⑥. The array can contain more, but must contain at least five tasks ⑦. If an error occurs when loading the tasks, the program can only be used in mode one.

④

⑤

⑥

⑦

```
[{
  "id": 0,
  "name": "task 0",
  "desc": "\\\"\\\"\\\"\\nTask: Create a function called add ... and returns the result!\\n\\\"\\\"\\\"\\\"",
  "code": "def add(a,b):\\n    pass",
  "test": "print((11 == add(5,6)) and (7 == add(4,3)))"
},{
  "id": 1,
  "name": "task 1",
  "desc": "\\\"\\\"\\\"\\\"\\nTask: Create a function called sub ... and returns the result!\\n\\\"\\\"\\\"\\\"",
  "code": "def sub(a,b):\\n    pass",
  "test": "print((11 == sub(22,11)) and (7 == sub(11,4)))"
},{
  "id": 2,
  "name": "task 2",
  "desc": "\\\"\\\"\\\"\\\"\\nTask: Create a function called mul ... and returns the result!\\n\\\"\\\"\\\"\\\"",
  "code": "def mul(a,b):\\n    pass",
  "test": "print((9 == mul(3,3)) and (21 == mul(7,3)))"
},{
  "id": 3,
  "name": "task 3",
  "desc": "\\\"\\\"\\\"\\\"\\nTask: Create a function called div ... and returns the result!\\n\\\"\\\"\\\"\\\"",
  "code": "def div(a,b):\\n    pass",
  "test": "print((11 == div(22,2)) and (7 == div(21,3)))"
},{
  "id": 4,
  "name": "task 4",
  "desc": "\\\"\\\"\\\"\\\"\\nTask: Create a function called mod ... and returns the result!!\\n\\\"\\\"\\\"\\\"",
  "code": "def mod(a,b):\\n    pass",
  "test": "print((1 == mod(3,2)) and (2 == mod(5,3)))"
}]
```

Have fun!