Documentation

Table of Contents

[1. TEchnical implementation 3](#_Toc89368436)

[1.1 ROLES 3](#_Toc89368437)

[1.2 Done work from anyone 3](#_Toc89368438)

[1.3 Formulas in the code 4](#_Toc89368439)

[2. program description 6](#_Toc89368440)

1. TEchnical implementation
   1. ROLES

|  |  |
| --- | --- |
|  |  |
|  | Alexander Baev – Backend Developer |
|  | Yusmen Osman – Backend Developer |
|  | Denis Kolev – Scrum Trainer |
|  | Kliment Tenev – QA |

* 1. Done work from anyone

|  |  |
| --- | --- |
|  |  |
| 1. 1 | Alexander Baev:  Formulas, documentation, and formulas comments |
|  | Yusmen Osman:  Formulas, connection in the main code and formulas comments |
|  | Denis Kolev:  Git Hub Repository, presentation, connection in the main code and code comments in the main |
|  | Kliment Tenev:  Formulas, presentation, and QA documentation |

### Formulas in the code

|  |  |
| --- | --- |
| № | Name of formula and what it does |
| 1 | weightOnOtherPlanets  Function that calculates how much a person weights on other planets |
| 2 | speedEqualIncrease  Function that calculates the final speed of a vehicle with a smaller starting speed |
| 3 | pathEqualIncrease  Function that calculates the distance from the moment of accelerating to the final speed |
| 4 | speedEqualDecrease  Function that calculates the final speed of a vehicle with a larger starting speed |
| 5 | pathEqualDecrease  Function that calculates the distance from the moment of accelerating to the final speed |
| 6 | stopTime  Function that calculates the time needed for a vehicle to stop |
| 7 | stoppingDistance  Function that calculates the stopping distance of a vehicle |
| 8 | freeFallingPath  Function that finds the path length using the free falling formula |
| 9 | freeFallingPathMercury, freeFallingPathVenus, freeFallingPathMars, freeFallingPathJupiter, freeFallingPathSaturn, freeFallingPathUranus, freeFallingPathNeptune, freeFallingPathPluto  Function that finds the path length on a specific planet using the free falling formula |
| 10 | freeFalling  Function that finds the speed using the free falling formula |
| 11 | freeFallingMercury, freeFallingVenus, freeFallingMars, freeFallingJupiter, freeFallingSaturn, freeFallingUranus, freeFallingNeptune, freeFallingPluto  Function that finds the speed on a specific planet using the free falling formula |
| 12 | freeFallingTime  Function that finds the time using the free falling formula |
| 13 | freeFallingTimeMercury, freeFallingTimeVenus, freeFallingTimeMars, freeFallingTimeJupiter, freeFallingTimeSaturn, freeFallingTimeUranus, freeFallingTimeNeptune, freeFallingTimePluto  Function that finds the time on a specific planet using the free falling formula |
| 14 | convertSecondsToMinutes  Function that converts seconds to minutes |
| 15 | convertMinutesToSeconds  Function that converts minutes to seconds |
| 16 | convertMsToKmh  Function that converts meters per second to kilometers per hour |
| 17 | convertKmhToMs  Function that converts kilometers per hour to meters per second |
| 18 | convertMetersToKilometers  Function that converts meters to kilometers |
| 19 | convertKilometersToMeters  Function that converts kilometers to meters |

## **program description**

When the program starts the user can choose from 5 categories of physics formulas by typing a specific number of each category. When this specific number is typed, the program shows all the formulas the user can choose from by typing a specific number or letter. After this happens the program tells the user what to input. After the user inputs the needed attributes, the program calculates the formula and shows it to the user. Finally, the program asks the user if he wants to continue or not. If he wants to continue the program starts again, otherwise it stops running.