Analysis and internal logistics (Backend) in the programming of the PlanetApp:

Objectives

To make use of specialized tools in Data Science to obtain reliable data, at an acceptable speed, and that adapt to the updates.

Used software:

- **Python 3.7.1**:

    - Pandas (Python Data Analysis Library)
    - NumPy (Powerful N-dimensional array object)
    - Sympy (Library for symbolic mathematics)
    - Requests (Elegant and simple HTTP library for Python)

Three modules are to be provided:

1. **exoplanets.py:** To receive, download/update and sort data in CSV format from NASA's exoplanet API.
2. **HelpBob.py:** Organizes progressive teaching didactic texts in the game of the virtual mascot "Bob", a pirate alien. Sort valuable information from exoplanets, such as their distance from our planetary system, to the type of star that orbits. And a list of planets that meet certain conditions returns: Distance, star type, star age [age of the planetary system], maximum mass, minimum mass, radius, and so on.
3. **limits.py:** Analysis of realistic limitations of mass, volume, density, zone of habitability. Tools for preventive analysis to avoid collisions in orbits.

exoplanets.py

The script has a dictionary with tuples of the different tables of Confirmed Planets, to make a lexical comparison, and thus correct the values that have been written incorrectly (if you ask 35 columns, you will not throw error for just one). This correction can be improved, but at least for most values it is functional.

The data update proceeds from "data.pickle" a file that has the data to download serialized.

The data currently used by the application is 35:

- **'pl_hostname', 'pl_letter', 'pl_name':** Star name, letter assigned to exoplanet, exoplanet name.
- **'pl_pnum':** Number of planet in the planetary system.
- **'pl_orbper', 'pl_orbsmax', 'pl_orbeccen':** Orbital period, semi-major axis and eccentricity.
- **'pl_bmassj':** Planet mass.
- **'pl_dens':** Planet density. (Few data)
- **'st_teff':** Effective or surface temperature of the star.
- '**st_mass':** Stellar mass.
- **'st_rad':** Stellar radius.
- **'pl_kepflag', 'pl_k2flag':** If the discovery is associated with Kepler 1 ó 2.

- **'st_dist':** Distance from the planetary system to ours.
- **'st_optmag':** Optical magnitude. (Later it will be more useful with the other bands, color data)
- **'pl_facility', 'pl_locale', 'pl_telescope', 'pl_instrument', 'pl_disc':** Name of facility of planet discovery observations, locale, telescope, instrument and year.
- **‘ra':** Right ascension. (It is intended to be used later for visualizations)
- **'pl_cbflag':**Its current use is to discard binary stellar systems. To avoid complications in the simulation. It could be integrated in the future.
- **'pl_eqt':** Equilibrium temperature or surface temperature of the planet. (Currently used in HelpBob explanations, but other geophysical and geochemical conditions of the planet could be used to give the real temperature, for a matter of time has not been given).
- **'pl_rade':** Planet radius.
- **'pl_trandur':** Transit Duration, can be used to explain astronomical measurements with light animations.
- **'pl_ratdor':** Planet-Star Distance over Star Radius. (To explain astronomical proportions)
- **'pl_mnum':** Moons.
- **'pl_pelink', 'pl_edelink':** Links of reference.
- **'st_lum':** Star luminosity. (Used for habitability zone calculations)
- **'st_dens':** Stellar density.
- **'st_age':** Star age.
- **'st_spstr':** Star type in Morgan-Keenan.
- **'st_umbj':** It is intended for use similar to **st_optmag**.

HelpBob.py

The filtering options are explained in GitHub. The dynamic texts used for learning in the App are:

```
>>> import HelpBob as hb
>>> a=hb.quest('whatexo')['exo_definition']
>>> print(a['title'], '\n\n', a['content'])
What is an exoplanet?

 Exoplanet - Any planet beyond our solar system

But we can ask ourselves, what is a planet?

A planet is an astronomical body that orbits a star (like our Earth around the Sun
).  Its massiveness gives it a rounded shape. And it is dominant in its orbit (the
 planet orbits "only" in its orbit).

I know concepts like "massiveness" can be a little strange. Even for physicists, t
his is why researching and digging into the essence of the universe is so intrigui
ng.

Never stop asking yourself questions!

The nearest exoplanet found is called Proxima Cen b and it's aprox. 1.3 parsecs aw
ay.
The farthest exoplanets found are called SWEEPS-11 b, SWEEPS-4 b and they're aprox
. 8500.0 parsecs away.

To travel to one parsec distance would take about 3.26 years at the speed of light
. Our fastest rocket built would take about 9651 years.
```

Source: https://exoplanets.nasa.gov/ , https://es.wikipedia.org/wiki/Planeta, https://es.wikipedia.org/wiki/Juno_(sonda_espacial)

```
>>> a=hb.quest('claexo')['exo_clasification']
>>> print(a['title'], '\n\n', a['content'])
Exoplanet naming convention

 For exoplanets orbiting a single star, the designation is normally formed by taki
ng the name or, more commonly, designation of its parent star and adding a lower c
ase letter.

"I understand, I understand, but how are stars named? Where do those fuzzy names c
ome from?"

You can ask me, and the answer is a little more confusing, it depends on the 'cata
log'.

The most common catalogs are:
Histoire CÃ©leste FranÃ§aise,  Bonner Durchmusterung,  Henry Draper Catalogue,  Br
ight Star Catalogue, and so on.

Some names are determined by the 'Mission' that found the star or that studies its
 stellar system frequently.
```

Source: https://en.wikipedia.org/wiki/Exoplanet_naming_convention

```
>>> a=hb.quest('teff')['temp_effectiva']
>>> print(a['title'], '\n\n', a['content'])
What is Effective Temperature?

 The effective temperature of a star indicates the amount of heat that the star ra
diates per unit of surface area. From the warmest surfaces to the coolest is the s
equence of stellar classifications known as O, B, A, F, G, K, M.

(We'll talk about the star rating soon)

In other words, the effective temperature is not the actual temperature of the sta
r, but a measurable parameter of its surface.

We must take into account that the stars are very far away and the effective tempe
rature is a good measure accessible for various calculations:

As the CHZ (Circumstellar Habitable Zone), do not be fooled by the name, the condi
tions for life to be possible, require more than this condition.

The star with the lowest Effective Temperature found is called WISEP J121756.91+16
2640.2 A and it's aprox. 575.0 Kelvins.
The star with the highest Effective Temperature found is called V0391 Peg and it's
 aprox. 29300.0 Kelvins.

The Kelvin is one of the most widely used temperature units in physics. Because co
nveniently its 0 is the absolute zero (there is no object colder than 0 Kelvin).

A cup of hot coffee has about 322 K, the effective temperature of the sun is 5778
k, 18 times the temperature of a coffee.

However, with 18 hot cups you could not reach the temperature of the sun, as this
is a statistical measure of heat. In addition, the sun in its deepest layers can r
each 15 million Kelvin.
```

Source: https://en.wikipedia.org/wiki/Planetary_equilibrium_temperature, https://en.wikipedia.org/wiki/Structure_of_the_Earth

```
>>> a=hb.quest('chz')['chz']
>>> print(a['title'], '\n\n', a['content'])
What is Circumstellar Habitable Zone (CHZ)?

 In popular jargon there is a tendency to think that the CHZ indicates the area in
 which a planet can possess life. This is not exactly the case.

The 'Habitable Zone' is the range of orbits around a star within which a planetary
 surface can support liquid water given sufficient atmospheric pressure.

In other words, if the planet has water, it can be in a liquid state. Fulfilling o
ne of the possible conditions for life. But not by guaranteeing it.

It is a necessary condition, not a sufficient condition.

There are mathematical approximations of the CHZ, but still remain in the theoreti
cal field, since we only have the certainty of the CHZ of our solar system.

Studies continue and we may someday be able to reformulate CHZ, providing a genera
l expression for life in planetary systems. And humanity would make a great leap i
nto interplanetary exploration.
```

Source: https://es.wikipedia.org/wiki/Zona_de_habitabilidad, https://en.wikipedia.org/wiki/Circumstellar_habitable_zone, https://exoplanetarchive.ipac.caltech.edu/docs/poet_calculations.html

```
>>> a=hb.quest('plsystem')['planet_system']
>>> print(a['title'], '\n\n', a['content'])
What is Planetary System?

 It's formed by a central star or several ( stellar system), and different objects
 orbiting around it. Our planetary system, the solar system, is formed by the Sun,
 the different planets and a multitude of minor bodies.
>>>
```

Source: https://en.wikipedia.org/wiki/Planetary_system, https://simple.wikipedia.org/wiki/Star_system

```
>>> a=hb.quest('clastar')['star_clasification']
>>> print(a['title'], '\n\n', a['content'])
How are stars categorized (Spectral Type)?

 Astronomers classify stars based on their spectral characteristics, which provide
 information about the temperature on their surfaces (Effective Temperature).

The method consists of analyzing the light of stars through diffraction grids. Inc
oming photons are divided into a spectrum of colors. Each of these values shows th
e presence of a chemical element, the excitation of which occurs at a certain degr
ee of temperature.

Some letters of the alphabet relative to each heat level have been used to make th
e stellar description: O, B, A, F, G, K, and M, in descending order. The stars O w
ould be the hottest, the M the coldest. But there is also a non-formal classificat
ion associating colors to these stars. So the stars O are called blue; B, blue-whi
te; A, white; F, yellow-white; G, yellow; K, orange, and M, red stars; although th
ese are not necessarily the visible tones.

The most up-to-date method for classifying stars is called the Morgan-Keenan syste
m. It adds a numerical range -between 0 and 9- between the spectra of each nominal
 letter. A5 would be 5 tenths between A0 and F0, the same principle of numerical f
ractionation, which allows for greater accuracy in spectral values.

Astronomers also include in this system the luminosity of stars, expressed by Roma
n numerals: I, II, III, IV and V. These values indicate the width of the absorptio
n bands of the spectrum, which in turn refers to the size of these stars. Class I
are the supergiants; class III, the giants; V are the dwarfs or main sequence star
s.

The spectrum of the Sun would be G2V, which would be read as a main sequence yello
w star, two tenths of the orange star.
```

Source: https://en.wikipedia.org/wiki/Stellar_classification, http://astronomy.swin.edu.au/cosmos/h/harvard+spectral+classification, http://astronomy.swin.edu.au/cosmos/H/Hertzsprung-Russell+Diagram, https://astro.unl.edu/naap/hr/hr_background1.html

```
>>> a=hb.quest('universal_age')['universal_age']
>>> print(a['title'], '\n\n', a['content'])
Ages and more ages:

 The universe is very old, about 13.7 Gyrs. It doesn't seem much, unless we notice
 that 1 Gyr is a billion years!

Our sun possesses 4.6 Gyr, 1/3 of the age of the universe, not bad.

The age of the galaxy is 13.51 Gyr, almost as much as the age of the universe.

And that of the earth is 4.5 Gyr, does not seem much compared to the sun, but 1000
00000 years difference is not a negligible figure.

Although it all depends on the scale and position from where you observe.
```

Source: https://en.wikipedia.org/wiki/Billion_years, https://en.wikipedia.org/wiki/Age_of_the_universe

There is mobile data, as is the case of the nearest and farthest planet. They are adapted based on what changes from the central data.

limits.py

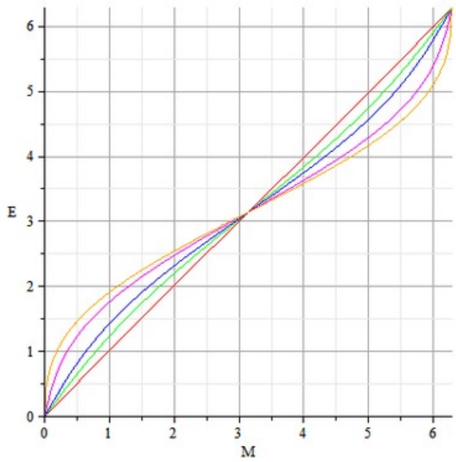
Mass and volume are limited based on the maximum and the minimum existing in Dataset. In the case of planets, the quotient between (mass of the planet)/(mass of the star) is compared, for each orbital system, the optimum quotient of comparison is obtained. Then the volume is related to the mass, and the maximum and minimum density. So that there is no possibility of simulating exaggerated volumes.

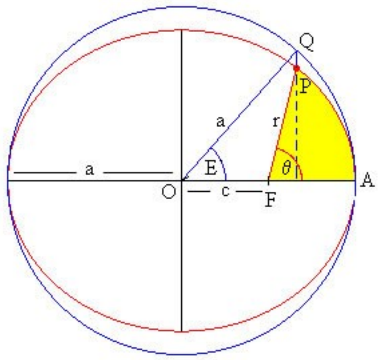
It also calculates density, this would be useful when simulating existing planetary systems and wanting to see the density of a planet that does not have it in the data table.

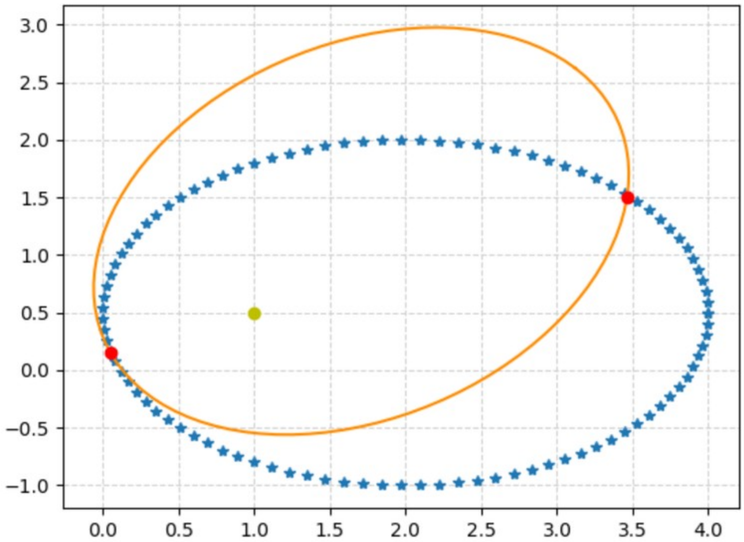
The calculation of the zone of habitability is a mathematical artifice, but it works to estimate, this is clarified in one of the HelpBob texts, to avoid the widespread confusion that "guarantees life". It also performed a function that calculates the luminosity of a star, based on temperature and radius of it.

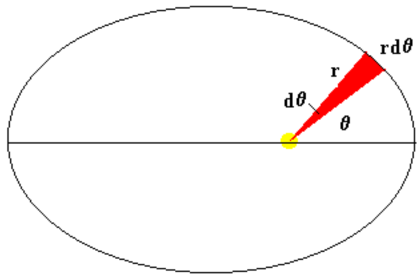
Finally, there is the collision module, the distance to the periastro, the tangential speed, and if necessary, the margin of "rigour". It processes two planets, but that's enough to evaluate it from the central program. In case of requiring indefinite planets. The use of Numpy makes it easier to make small modifications if required.

The mathematical and physical concepts used in this last case (the first two are solved with statistical management). It is a frequent use of Kepler's laws (and its transcendental equation), Newton's laws for gravitation (angular momentum and mechanical energy are conserved in elliptical orbits), knowledge of the relationship between periastro, apoastro and velocity, with eccentricity, semi-major axis and semi-major axis. Knowing that the central star is always in focus.


E
M


Q
P
a
r
E
θ
a
O
c
F
A




rdθ
r
dθ
θ