



All Code Clubs must be registered. By registering your club we can measure our impact, and we can continue to provide free resources that help children learn to code. You can register your club at codeclubworld.org.

Introduction

In this project, we'll be creating an interface to provide information on the USA and Russian moon missions. This will allow the user to select a date of a mission, and choose whether to see:

- Spacecraft used
- Operator
- Type of Mission

You will need Python version 3 for this project. Select Python3 in trinket.io or make sure you have Python3 installed on your computer.

```
Enter a date e.g. 21 February 1969,  
or 'q' to quit : 21 February 1969  
Do you want to the Spacecraft(1),  
Operator(2), or Mission(3) for that  
year? 1  
Do you want to search USA(u) or  
Russian(r) missions? r  
In 21 February 1969 the Russian  
Spacecraft was: 7K-L1S No.3  
-----  
Enter a date e.g. 21 February 1969,  
or 'q' to quit : █
```



Activity Checklist

Follow these **INSTRUCTIONS** one by one



Test your Project

Click on the green flag to **TEST** your code



Save your Project

Make sure to **SAVE** your work now

Step 1: Get the data

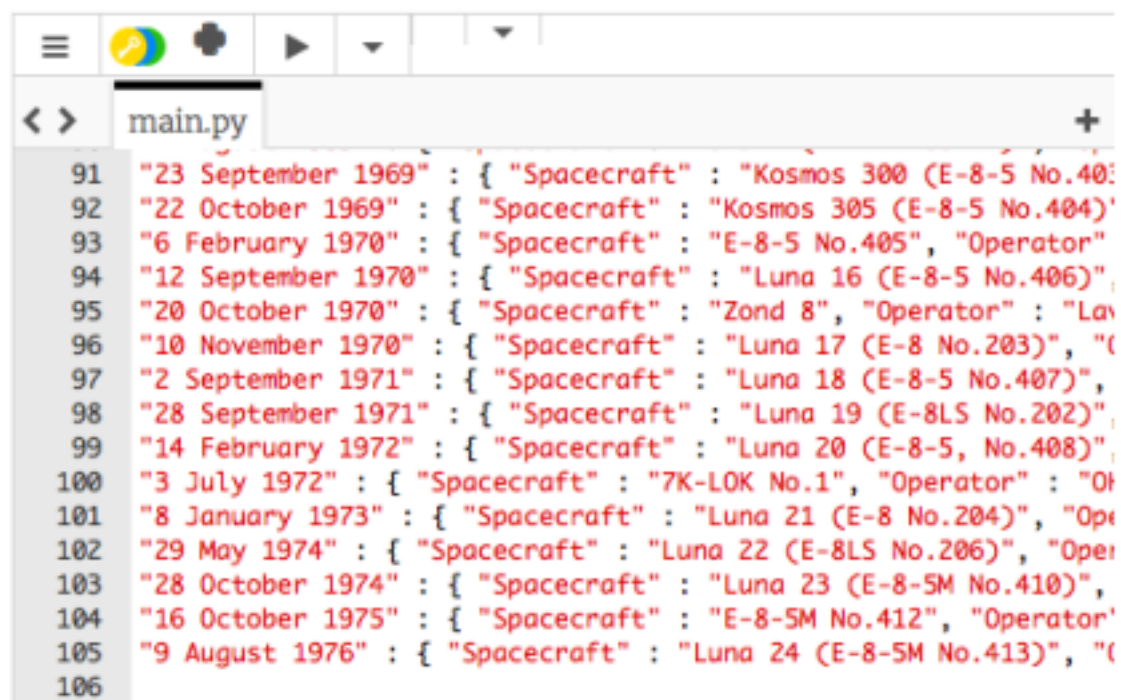
Go here to get the moonhack.py file. We've already added the data to start you off. Open the file in your IDLE editor, or a text editor like atom editor (from <https://Atom.io>), or notepad on Windows. You can also paste this from notepad into your trinket.io editor.

✓ Activity Checklist

- You'll see the text we've already typed in for you (phew!).



Python 3 Now Available! Learn More



```
91 "23 September 1969" : { "Spacecraft" : "Kosmos 300 (E-8-5 No.403)", "Operator" : "USSR" }
92 "22 October 1969" : { "Spacecraft" : "Kosmos 305 (E-8-5 No.404)", "Operator" : "USSR" }
93 "6 February 1970" : { "Spacecraft" : "E-8-5 No.405", "Operator" : "USSR" }
94 "12 September 1970" : { "Spacecraft" : "Luna 16 (E-8-5 No.406)", "Operator" : "USSR" }
95 "20 October 1970" : { "Spacecraft" : "Zond 8", "Operator" : "USSR" }
96 "10 November 1970" : { "Spacecraft" : "Luna 17 (E-8 No.203)", "Operator" : "USSR" }
97 "2 September 1971" : { "Spacecraft" : "Luna 18 (E-8-5 No.407)", "Operator" : "USSR" }
98 "28 September 1971" : { "Spacecraft" : "Luna 19 (E-8LS No.202)", "Operator" : "USSR" }
99 "14 February 1972" : { "Spacecraft" : "Luna 20 (E-8-5, No.408)", "Operator" : "USSR" }
100 "3 July 1972" : { "Spacecraft" : "7K-LOK No.1", "Operator" : "USSR" }
101 "8 January 1973" : { "Spacecraft" : "Luna 21 (E-8 No.204)", "Operator" : "USSR" }
102 "29 May 1974" : { "Spacecraft" : "Luna 22 (E-8LS No.206)", "Operator" : "USSR" }
103 "28 October 1974" : { "Spacecraft" : "Luna 23 (E-8-5M No.410)", "Operator" : "USSR" }
104 "16 October 1975" : { "Spacecraft" : "E-8-5M No.412", "Operator" : "USSR" }
105 "9 August 1976" : { "Spacecraft" : "Luna 24 (E-8-5M No.413)", "Operator" : "USSR" }
106
```

Step 2: Set up our main() function

- Directly underneath that is where we start coding. Let's define our `main()` function and start with a dictionary that maps our options to numbers.

```
def main():
    options = {
```

```
"1" : "Spacecraft",
"2" : "Operator",
"3" : "Mission"
}
```

Step 3: Get information for our query

- Since we might want to query multiple moon missions, let's put our code in a loop. Add the following lines to our code (we've shown the previous lines too, so you can see how it lines up).

```
def main():
    options = {
        "1" : "Spacecraft",
        "2" : "Operator",
        "3" : "Mission"
    }
    while True:
        date_to_search = input("Enter a date e.g. 21 February 1969\
, or 'q' to quit :")
        if date_to_search == 'q':
            return
        detail = input("Do you want to the Spacecraft(1)\
, Operator(2), or Mission(3) for that year?")
        country = input("Do you want to search USA(u) or Russian(r)\
missions? ")
```

These ask the user what they want information on. We've also included an `if` statement in case we're finished and want to quit.

🚩 Test your project

Run your code, and you should see the prompt asking for information.

```
Enter a date e.g. 21 February 1969,  
or 'q' to quit : █
```



Save your project

If you're using Trinket, you don't have to worry about this. If you're using an editor, make sure you save your file now. You can call it anything that ends in `.py`.

Step 4: Get the information

Remember the data we typed in for you before? That data is in what we call a dictionary. A dictionary has keys associated to values. In this case, the keys are dates, and the value is actually a mini dictionary that contains Spacecraft, Operator, and Mission.

✓ Activity Checklist

- To get this information from the `usa_missions` dictionary, we can use this format (you can try this in an interactive shell if you have it, but don't type it into our program just yet):

```
usa_missions['17 August 1958']
```

- This will result in: `{ "Spacecraft" : "Able 1", "Operator" : "USAF", "Mission" : "Orbiter" }`
- We can use the information from our `options` dictionary, and

the choice made by the user `detail`, to get just the Spacecraft used on that mission.

```
usa_missions['17 August 1958'][options[detail]]
```

This will just result in `Able 1` being displayed.

Step 5: Display the information nicely

- So now we can add this to our program. Here's the whole program (remember the data is above it in your file, we've just left it out for clarity).

```
def main():
    options = {
        "1" : "Spacecraft",
        "2" : "Operator",
        "3" : "Mission"
    }

    while True:
        date_to_search = input("Enter a date e.g. 21 February 1969, \
or 'q' to quit :")
        if date_to_search == 'q':
            return
        detail = input("Do you want to the Spacecraft(1), \
Operator(2), or Mission(3) for that year?")
        country = input("Do you want to search USA(u) or Russian(r)\
missions? ")

        if country == 'u' or country == 'r':
            if country == 'u':
                try:
                    print("In " + date_to_search + " the USA " \
+ options[detail] + " was: " + \
usa_missions[date_to_search][options[detail]])
                except:
```

```

        print("Sorry that was not found")
    if country == 'r':
        try:
            print("In " + date_to_search + " the Russian "\
+ options[detail] + " was: " +\
soviet_union_missions[date_to_search]
[options[detail]])
        except:
            print("Sorry that was not found")
    print("-----")
main()

```

At the end, we call the `main()` function that we've defined.

- If there was no mission the date entered, or incorrect information was entered, our program will print out 'Sorry that was not found'. This is why we put the `try:` and `except:` sections in our code. After the information is displayed to the user, it will go back to the start of our loop.

```

Enter a date e.g. 21 February 1969,
or 'q' to quit : 21 February 1969
Do you want to the Spacecraft(1),
Operator(2), or Mission(3) for that
year? 2
Do you want to search USA(u) or
Russian(r) missions? r
Sorry that was not found
-----
Enter a date e.g. 21 February 1969,
or 'q' to quit : █

```

Test your project

Try running your project and make sure it works. If it doesn't work, check that the indentation is exactly as it is in the above listing.



Save your project

Challenge: Challenge name

- Give the user the choice to list all the dates
- Give the user the choice to list all details for a mission.
- Add a moon-inspired ascii art banner when the program starts.