

Recap some of the Dict tricks

```

dat = {'CIE': 1, 'APH': 2, 'JSPS': 3 }
                                # Assign a dict key/value pairs
dat = {}                        # Empty dict
dat['CIE']                       # Access a value of the given key
dat.keys()                       # Get all keys
dat['JAM'] = 4                   # update value if the given key exists
                                # otherwise, add a new key/value pair
del dat['CIE']                   # Remove an item of the given key
dir(dat)                         # See all attributes and methods

```

!Caution! similar to a list assignment, a dict assignment is a reference assignment, not a new copy, i.e., `d1 = dat` is different from `d2 = dat.copy()`.

***** Note: The order of a dictionary may seem strange. Do not worry about it. Dictionary emphasizes access using keys, rather than an order.**

Don't worry if you don't get all the tricks, just have fun playing with them.

=====

Use `if __name__ == '__main__':` for better code organization and allowing smooth autograding.

P1. Write a function named `make_dict` taking no argument, creating a dict having key-value pairs of 'H': 1, 'He': 2, 'C': 6, 'N': 7, 'O': 8, and returning the created dict.

Example

When invoked by

```
atomic = make_dict()
```

```
print(atomic)
```

it results

```
=====
{'He': 2, 'H': 1, 'N': 7, 'C': 6, 'O': 8}
=====
```