SimCity Land Value Calculator Version 1: Read and display land values

In this version, you will read the data from a file and display the land values as a grid.

What to do

- First, study the format of the data files. The first line of the file contains the number of rows in the grid, and the second line contains the number of columns in the grid. The remaining lines contain the land values in the grid.
- Implement the `create_grid` function that creates the grid by reading information from a file.
- Implement the `display_grid` takes the grid in its parameter and displays the grid. The values in each column of the grid should be displayed right-justified within 9 spaces.
- You must use the given template means you should not change the names of the functions, its parameters or the object it returns.

You must use the given template:

```
def create_grid(filename: str) -> list[list[int]]:
    """
    Create a grid of land values from a file
    """
    # TODO: Implement this function
    pass

def display_grid(grid: list[list[int]]) -> None:
    """
    Display a grid of land values
    """
    # TODO: Implement this function
    pass

def main() -> None:
    """
    Main program.
```

```
    """
    grid = create_grid("data_0.txt")
    print("SimCity Land Values:")
    display_grid(grid)
```

Hints

- Outer `for` loop would iterate through rows while the inner `for` loop would iterate through columns
- Values are separated in the file by `\n` (end of line) character
- Study the following example to understand how values can be displayed within the specified column width. This information can be used to implement the `display_grid` function

```
>>> # Objects of type str are left justified within 10 spaces
>>> a = "dog"
>>> b = "cat"
>>> c = "mouse"
>>> print(f"{a:10} {b:10} {c:10}")
dog         cat         mouse
>>> # Objects of type int are right justified within 10 spaces
>>> a = 200
>>> b = 1656
>>> c = 34
>>> print(f'{a:10}{b:10}{c:10}')
       200      1656        34
```

Program name

Save your program as `simcity1.py`.

Demo

In this demo, `data_1.txt` is used.

https://asciinema.org/a/WpvfVDba3rm4FE7fGLl0SGyVF

Testing

To make sure your program works correctly, you should test it.

- Run your program with `python simcity1.py` with `data_0.txt`. Your program should print:

```
Sim City Land Values:
       1        0        3        4
       5        6        7        8
       9       10       11       12
      13       14       15       16
```

- Run your program with `python simcity1.py` with `data_1.txt`. Your program should print:

```
Sim City Land Values:
   76000        0    54000    16000    83000
   27000    49000    62000        0    31000
       0    48000    53000    22000    19000
   71000    37000    63000    41000        0
   83000    25000        0    16000    59000
```

- Run your program with `python simcity1.py` with `data_2.txt`. Your program should print:

```
Sim City Land Values:
   94000    64000    30000        0    14000    92000
   37000    49000    50000    29000    35000        0
       0    88000    85000    96000    60000    22000
   13000    44000    73000        0    45000    53000
   20000    33000    67000    71000    82000        0
   36000        0    62000    55000    44000    75000
```

- Run your program with `python simcity1.py` with `data_3.txt`. Your program should print:

```
Sim City Land Values:
   24000    57000    50000    43000
```

```
   38000        0    16000    62000
   51000    25000    49000        0
       0    76000    19000    34000
```

Submitting

Submit `simcity1.py` via eClass.