1. get_usa_data()

Retrieves usa data from nyt git repo (csv), using pandas
Sets values for:

1. usa_increase_deaths
2. usa_increase_cases
3. curr_date
4. usa_total_cases
5. usa_total_deaths

returns DataFrame with columns listed below

usa_data - DataFrame

Index	date	cases	deaths	case_increase	death_increase
0	2020-01-21	1	0	nan	nan
1	2020-01-22	1	0	0	0

2. get_states_data(state, starting_date)

Retrieves states data from nyt git repo (csv), using pandas
Sets values for:

1. st_increase_deaths
2. st_increase_cases
3. curr_date

returns DataFrame with columns listed below, for state in parameter

stdata - DataFrame

Index	date	state	fips	cases	deaths	case_increase	death_increase
241	2020-03-01	California	6	33	0	5	0

3. get_max_increase(category)

Retrieves states data from nyt git repo (csv), using pandas

returns DataFrame with columns listed below, one row for each state,for today's date, sorted by max cases or max death.

max_cases - DataFrame

Index	date	state	fips	cases	deaths	deaths_increase	cases_increase
0	2020-08-10	California	6	574267	10460	95	11023
1	2020-08-10	Texas	48	514940	9058	80	6750
2	2020-08-10	Florida	12	536953	8276	91	4155

4. create_states2_chart(my_data)

Calls get_states_data(state, st_date). Retrieves df for state passed.
Sets values for:

graphJSON_states2_cases
graphJSON_states2_deaths

Returns a JSON string which is then used to build the plotly chart.
String represents data used to build the chart for one state.
String includes x axis values, y axis values, color, opacity.

```
@app.route('/get_usa_chart/')
5.    create_usa_chart(my_data)
```
Calls get_usa_data()
Sets values for:
- graphJSON_usa_cases
- graphJSON_usa_deaths
- usa_increase_deaths
- usa_increase_cases

Returns the rendered usa.html

```
@app.route('/get_max/<category>')
6.    get_max(category)
```
Calls get_max_increase(category)
Sets values for:
- data_cases (this is a list of graphJSON objects)
- data_deaths
- max_cases
- max_deaths
- states

Returns rendered top_states2_html.html

```
@app.route('/get_state', methods=['POST'])
7.    get_state()
```
Calls these functions:
- get_states_data()
- create_states2_chart()

Sets values for:
- user_state (from form)

Returns rendered select_state_html.html

```
@app.route('/form/<category>', methods=['GET', 'POST'])
8.    form(category)
```
Calls these functions:
- get_max(category)

Sets values for:
- top (from form)
- starting_date (from form)

Returns rendered top_states2.html

```
@app.route('/')
9.    def index():
```
Calls get_usa_data() to set values for:
- usa_total_deaths
- usa_total_cases
- curr_date

Returns rendered index.html