73RD SCIENCE TALENT SEARCH

Project Title:

Simulation of an epidemic spread with Python

By Leo Guan (4S, Livingstone PS)
2024

Introduction

Name: Leo Guan

Age: 10 Years old

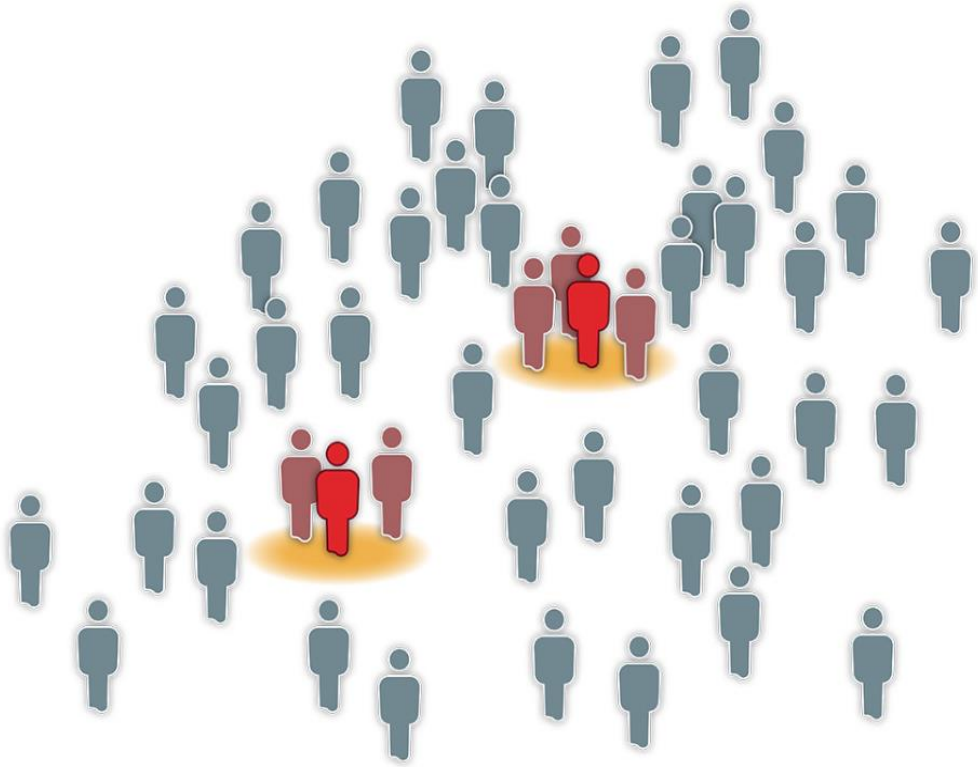
Grade: 4S

School: Livingstone PS

I am a happy boy and interested in computer science.

Why I chose “Simulation of an epidemic spread” as project

- Topic for 2024: Species Survival – more than just sustainability



- Viruses, bacteria, and other pathogens can cause disease outbreaks and epidemics.
- The COVID-19 pandemic has led to a dramatic loss of human life worldwide and presents a large challenge to public health, food systems and the world of work.

World Health Organization

Target for a Simulation of an epidemic spread

- helping us understand better the spread of diseases
- What are the factors responsible for the establishment of an epidemic?
- make more efficient and more effective ways to stop diseases
- Make the prediction about the development of diseases

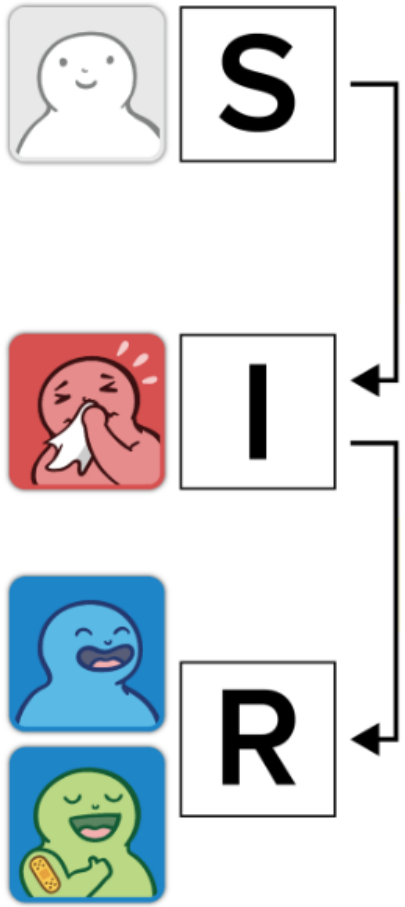
What is “SIR Model （ **S**usceptible, **I**nfectious, **R**emoved ）”

The SIR model is a simple, most-used mathematical model of epidemics spread.

The SIR model is dynamic, meaning that the number of individuals in each group changes over time. You can use the model to predict the number of susceptible, infectious, and removed individuals in a population at specific timepoints.

some assumptions of my version of the SIR model:

- The population is divided into three distinct groups: susceptible people, infectious people, and removed people.
- The total population never increases or decreases.
- When infectious and susceptible people interact the susceptible will become infected, infected people can start infecting immediately
- Once someone recovers, they are immune and cannot become reinfected.
- the infection rate is fixed.



Susceptible
Uninfected and unvaccinated individuals who can become infected

Transmission
The process by which susceptible individuals become infected

Infectious
Individuals who are infected with a given pathogen and can pass the pathogen to susceptible individuals

Recovery
The process by which infected individuals recover

Removed
1. Individuals who were infected but have recovered, and are now immune (cannot be reinfected)
2. Individuals who are vaccinated and are now immune

Important Formulation

$$\frac{dS}{dt} = -\beta SI$$

$$\frac{dI}{dt} = \beta SI - \gamma I$$

$$\frac{dR}{dt} = \gamma I$$

Susceptible

$$S_{i+1} = S_i - (\beta S_i I_i)dt$$

Infectious

$$I_{i+1} = I_i + (\beta S_i I_i - \gamma I_i)dt$$

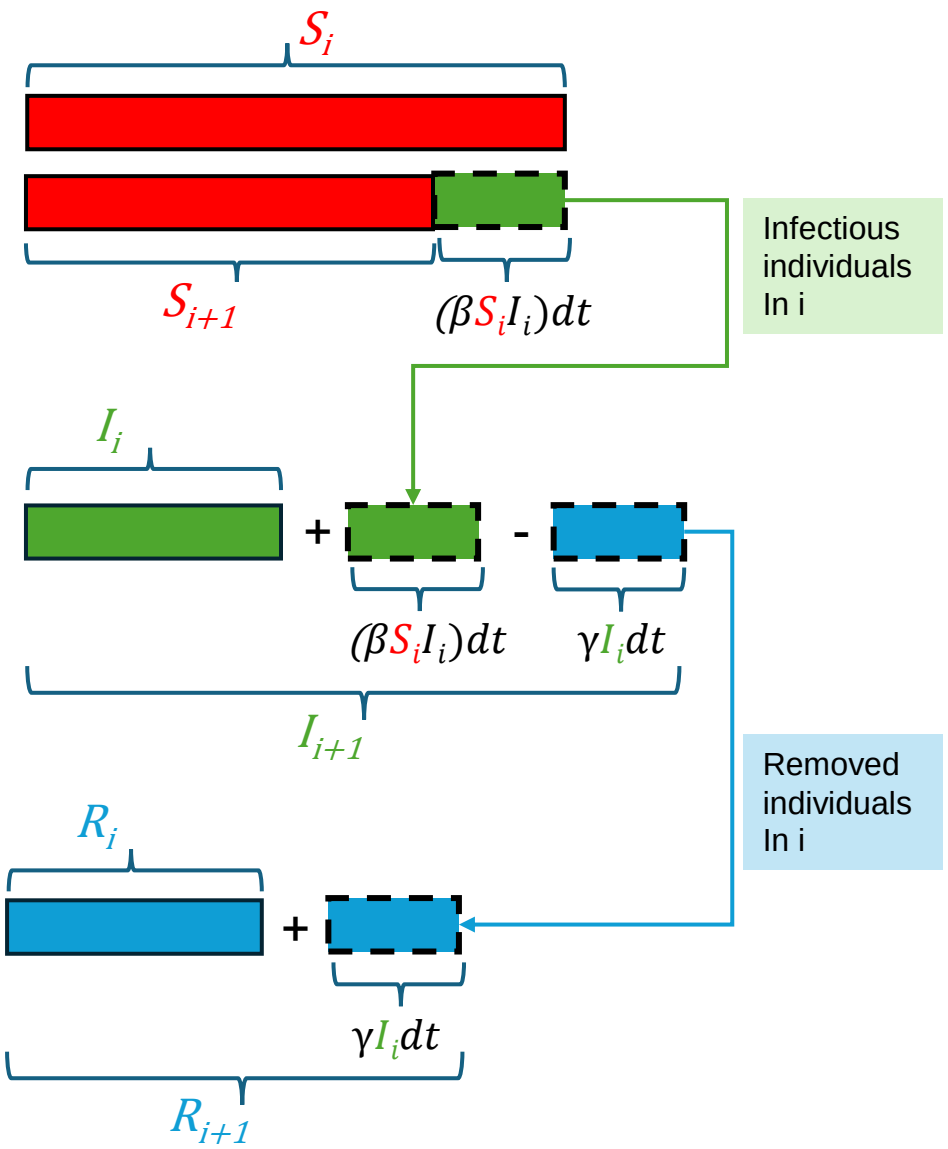
Removed

$$R_{i+1} = R_i + \gamma I_i dt$$

$\boldsymbol{\beta}$ = contacts per infected per day

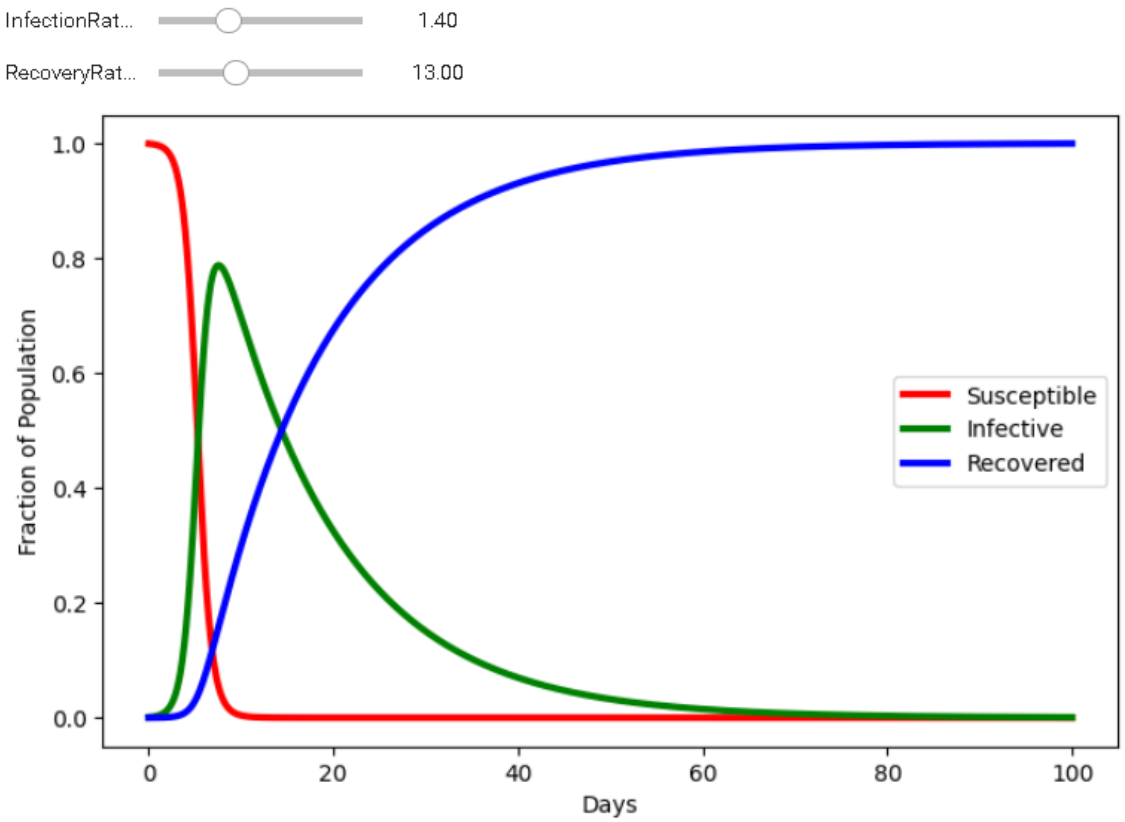
$\boldsymbol{\gamma}$ = recoveries per *person per day*

$\frac{\mathbf{1}}{\boldsymbol{\gamma}}$ = days to recover



Illustrations not to scale

What does my Program do?




```
# Part 4 - Output of Results
print("The maximal infected Population = ");print(I.max())
```

```
The maximal infected Population =
0.7873583684356391
```

My program simulate an infectious disease outbreak in a small population, collect data, and generate a graph.

Important data:

- susceptible
- infectious
- Recovered (removed)
- Infection Rate β
- Recovery Rate γ

Important result:
the maximal infected population

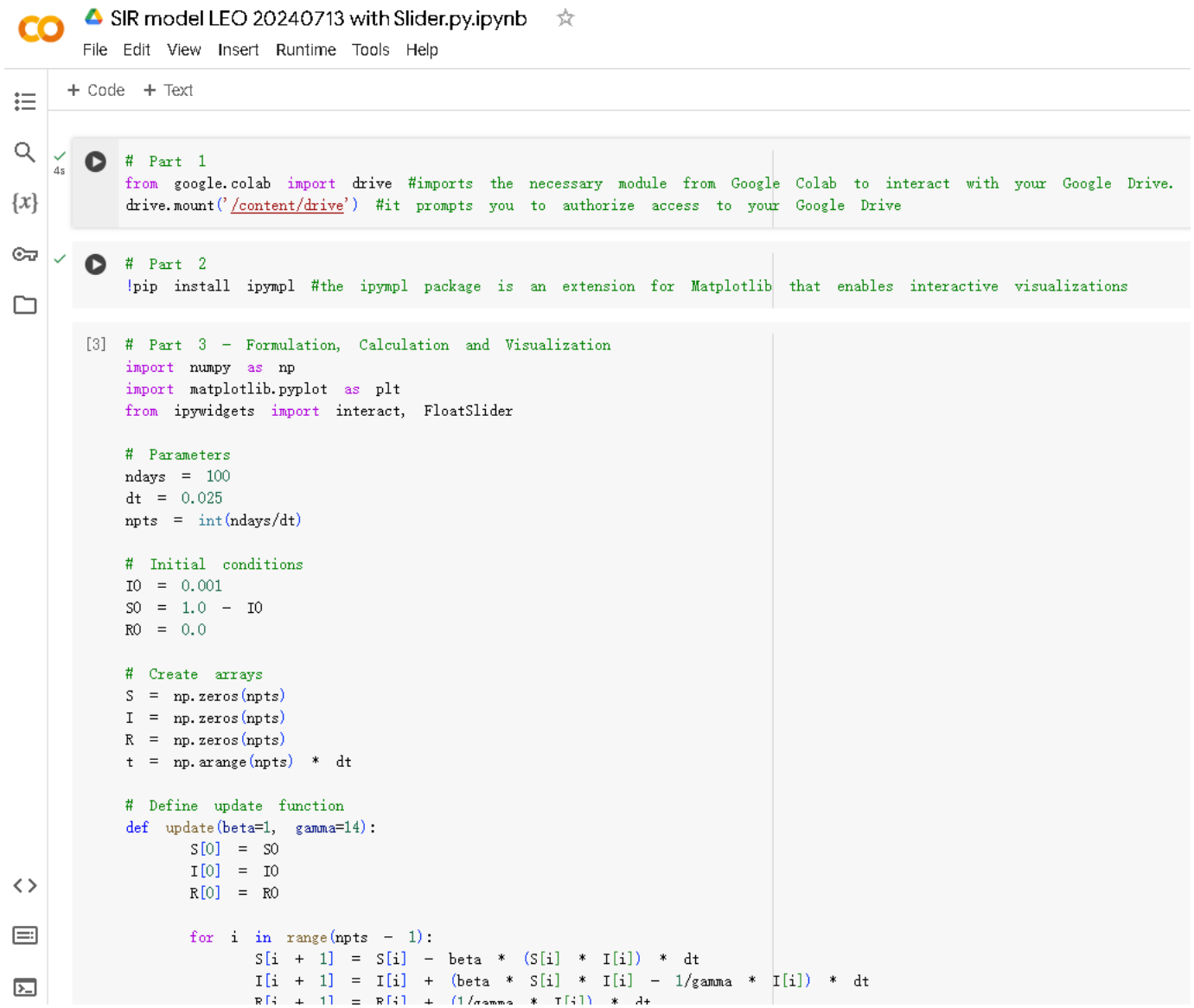
How to go to Collaboratory(where my code runs)

- **Programming Language** used: Python
- **Operating System** used: Windows

- You can run my program also online by Google Collaboratory

Google colab

→ Just double click my program link
https://colab.research.google.com/drive/1HtHj0DUH0NLgMjwX8CH_AkC9M-qB1LX2?usp=sharing
→then following the instruction of Collaboratory
(it is necessary that you have a Goggle Account)
→ And now you get ready to start.



How to run my program

There are 4 Parts of the program, that you should run

- ☐ Part 1: prepare the necessary module for authorization

Click the button ▶ and get this message `Mounted at /content/drive`

- ☐ Part 2: prepare the necessary module for Visualizations.

Click the button ▶ and get this message `Successfully installed ipympl-0.9.4 jedi-0.19.1`

- ❑ Part 3: Calculation and Visualization.

Click the button ▶ and get the initial graph (see Pic. 1)

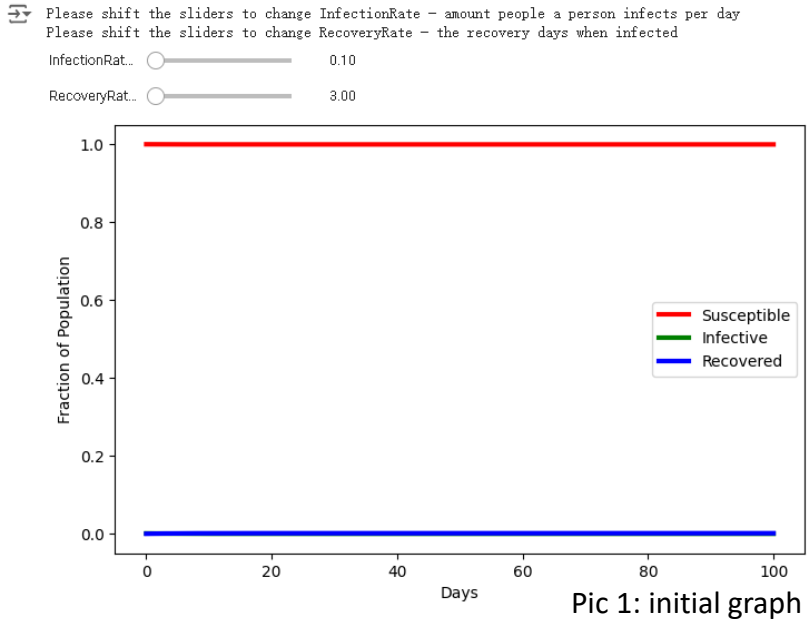
→shift the sliders and observe the changes (see Pic 2 for e.g)

- ☐ Part 4: Results

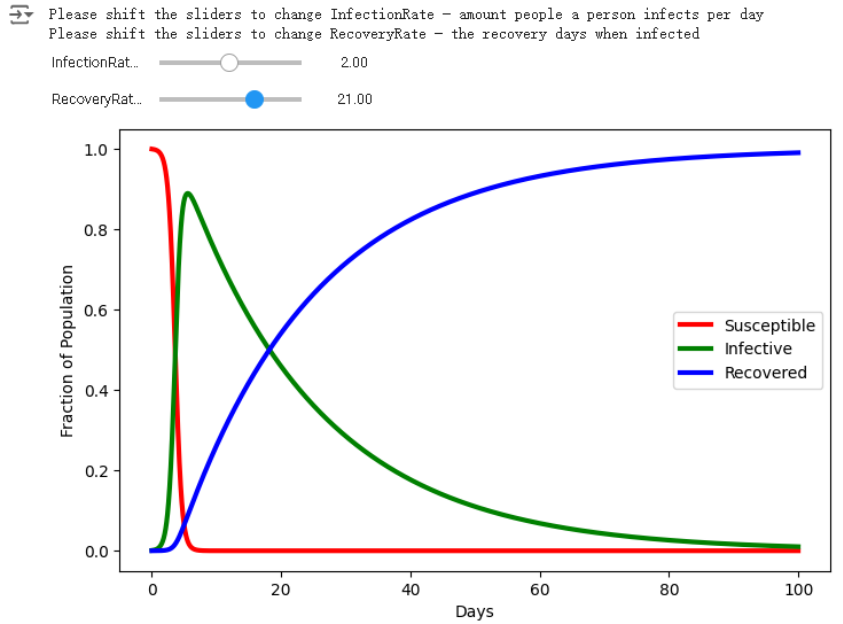
Click the button ▶ and get the result (maximal infected Population)

```
The maximal infected Population =
0.888778547645114
```

It means that the 88.88% of population will be infected if Infective Rate =2 and Recovery Rate =21days.



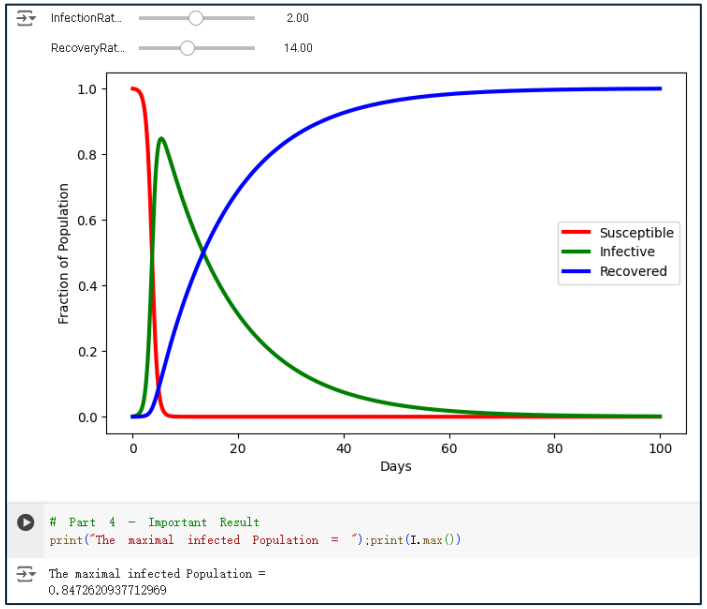
Pic 1: initial graph

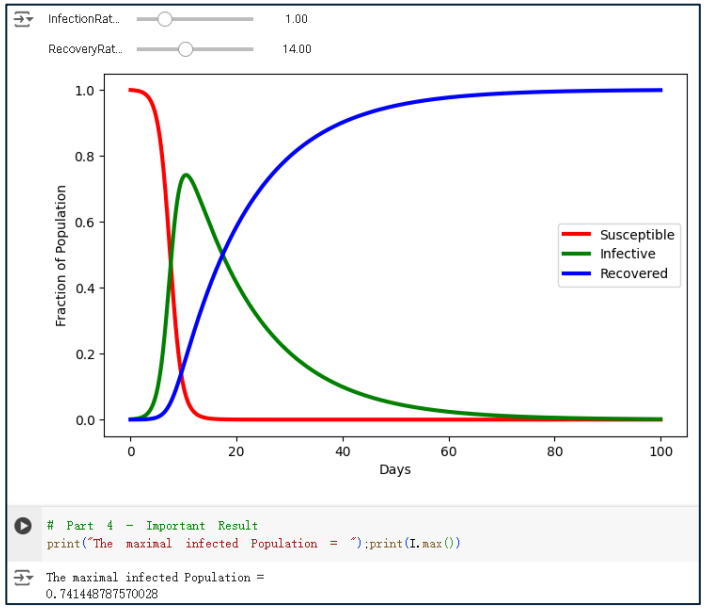


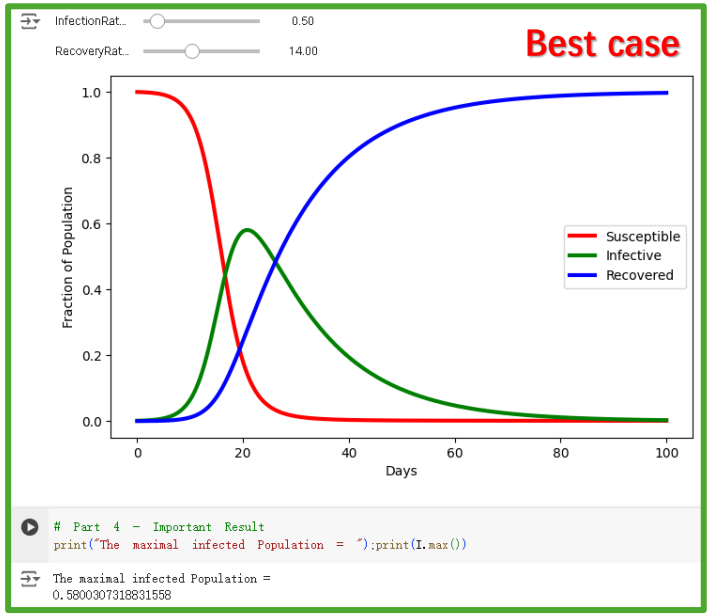
Pic 2: modified graph

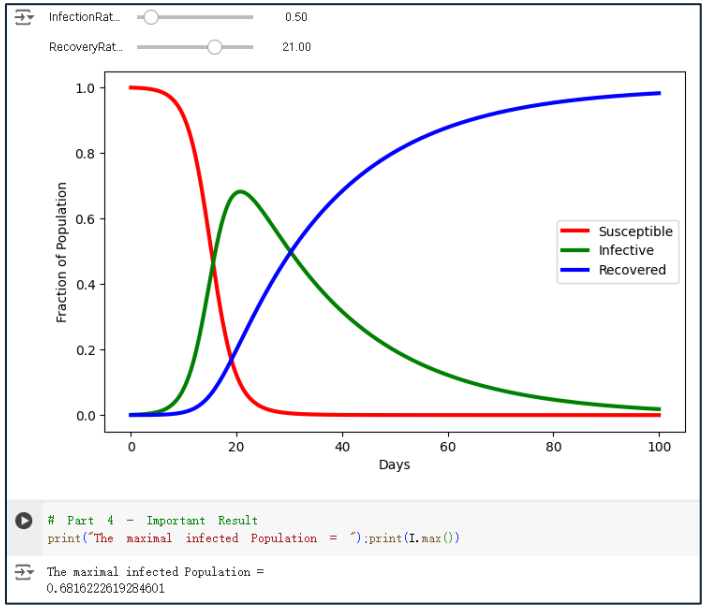
What does it mean? - Analyze

To test the program, we try to input different values to see how the epidemic will spread.









The results show that the best case of infected population is, with the lowest infection rate and recovery rate.

	Infection Rate	Recovery Rate	maximal infected population
1	2	14	0.8473
2	1	14	0.7414
3	**0.5**	**14**	**0.58**
4	0.5	21	0.6816

Our target should be to keep the infection rate/recovery rate very low and the “Green line – infected” as flat as possible, so that the maximal infected population low as possible.

That is why the government forced things like wearing masks, respecting social distancing, total or partial lockdowns and quarantine.

What do I want to optimize with this program?

- **More parameters (factors) for Beta like Social distance, lockdown, etc.…**
- **Better visual presentation**
- **User friendly and interactive things**

It’s my first Python program, I will keep learning my lessons for Python