
20ml
CORONAVIRUS
COVID-19
VACCINE
INJECTION ONLY
COVID -19 VACCINATION
Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed
diam nonummy nibh euismod tincidunt ut laoreet
shutterstock
IMAGE ID: 1894960762
www.shutterstock.com

- *Creating a COVID-19 vaccine analysis program would typically involve data analysis and visualization. Below are examples of Python code snippets using popular libraries like NumPy, Pandas, and Matplotlib to get you started.*

1.Data Retrieval:

You can fetch COVID-19 vaccine data from sources like the CDC or WHO, or use publicly available datasets. Here's an example using Pandas to read a CSV file:

- import pandas as pd# Read vaccine data from a CSV filevaccine_data = pd.read_csv('vaccine_data.csv')

2.Data Exploration:

You might want to explore the dataset by checking for missing values, data types, and basic statistics:

```
# Check for missing values
missing_values =
 vaccine_data.isnull().sum()
# Get basic statistics
summary_stats = vaccine_data.describe()
```

- *3.Data Visualization:*
- *Visualize vaccine distribution or coverage using Matplotlib or another plotting library:*

```
import matplotlib.pyplot as plt

# Plot vaccine coverage over time
plt.figure(figsize=(10, 6))
plt.plot(vaccine_data['Date'], vaccine_data['Coverage’
])
plt.xlabel('Date')plt.ylabel('Vaccine Coverage
(%)')plt.title('COVID-19 Vaccine Coverage Over
```

- **4.Data Analysis:**
- You can perform specific analyses, like calculating the average vaccine coverage or identifying regions with lower coverage:

```
# Calculate average vaccine
 coverage
avg_coverage =
 vaccine_data['Coverage'].mean()
# Find regions with lower coverage
low_coverage_regions =
vaccine_data[vaccine_data['Coverage'] < 50]
```

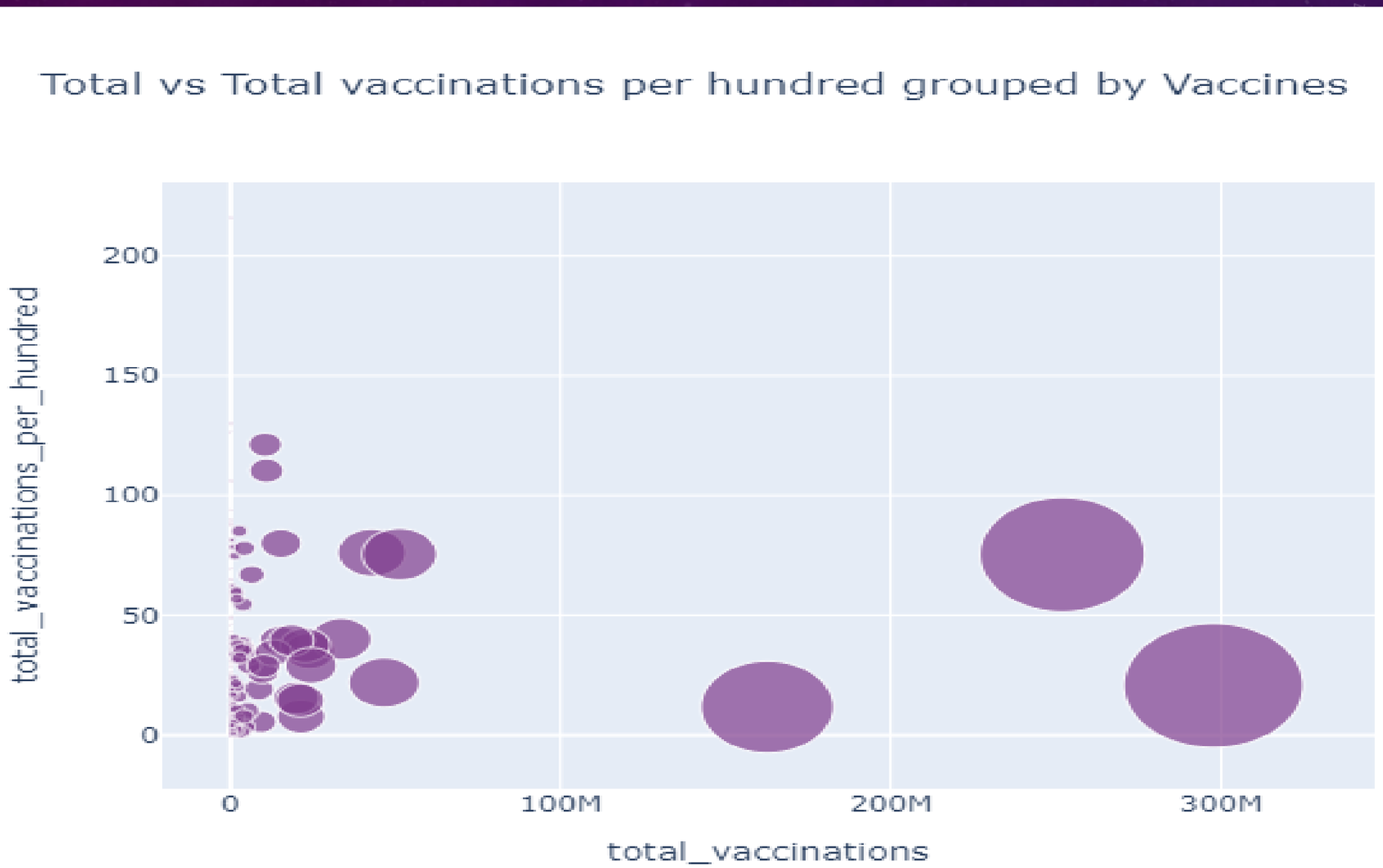
- ***5.Machine Learning (Optional):***
- *If you have enough data, you can build predictive models. Here's a simple linear regression example using scikit-learn:*

```
from sklearn.model_selection import
train_test_split
from sklearn.linear_model import
LinearRegression
```

- *# Split the data into training and testing sets*
- *X = vaccine_data[['Population']]*
- *y = vaccine_data['Coverage']*
- *X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)*
- *# Create and train a linear*
- *regression model*
- *model = LinearRegression()model.fit(X_train, y_train)*

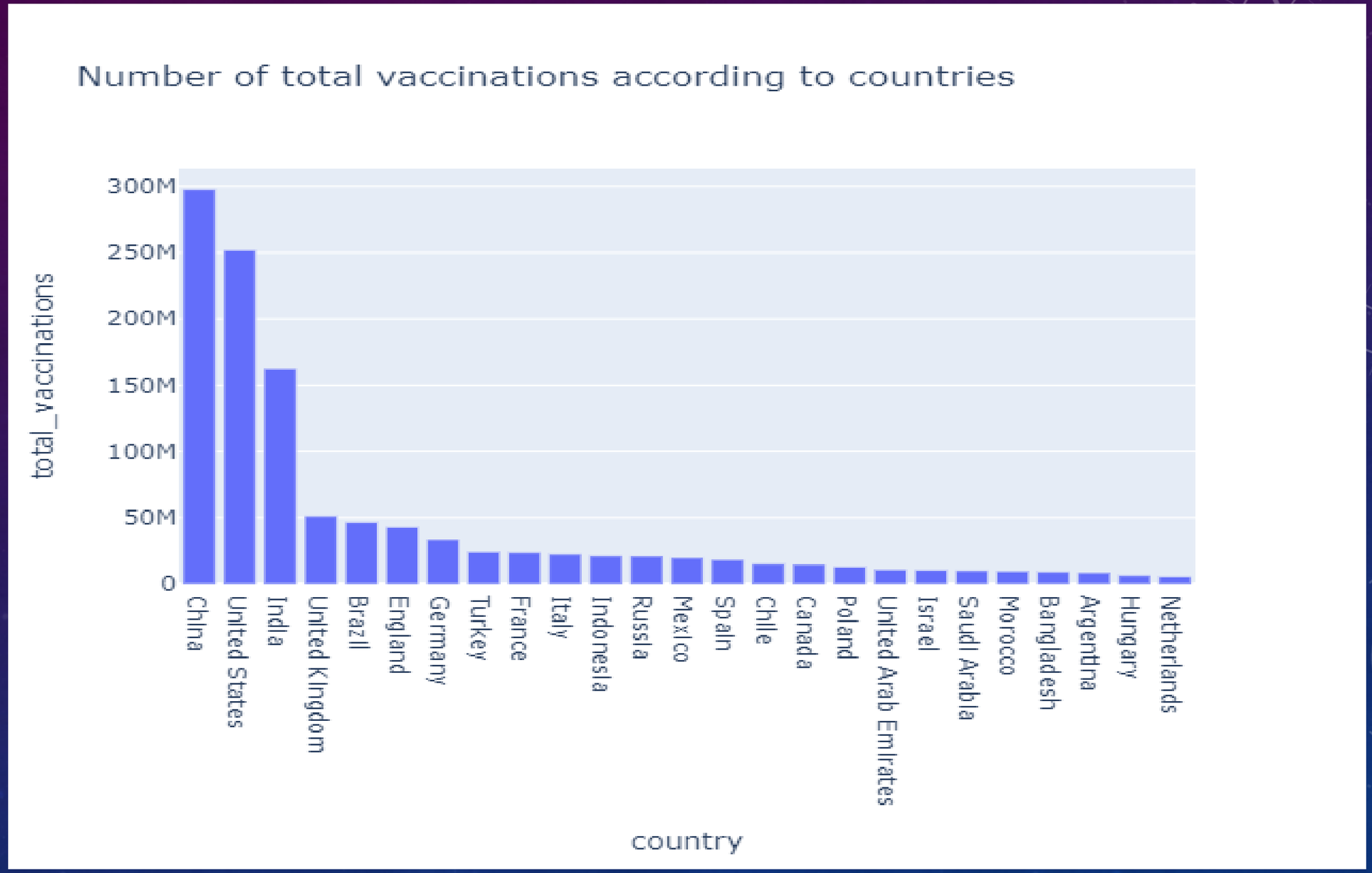
- ***Relation between Total Vaccinations and Total Vaccinations per Hundred:***

```
fig = px.scatter(new_df,x = 'total_vaccinations',y='total_vaccinations_per_hundred', size='total_vaccinations', hover_name = 'country',size_max = 50, title="Total vs Total vaccinations per hundred grouped by Vaccines", color_discrete_sequence = px.colors.qualitative.Bold) fig.show( )
```

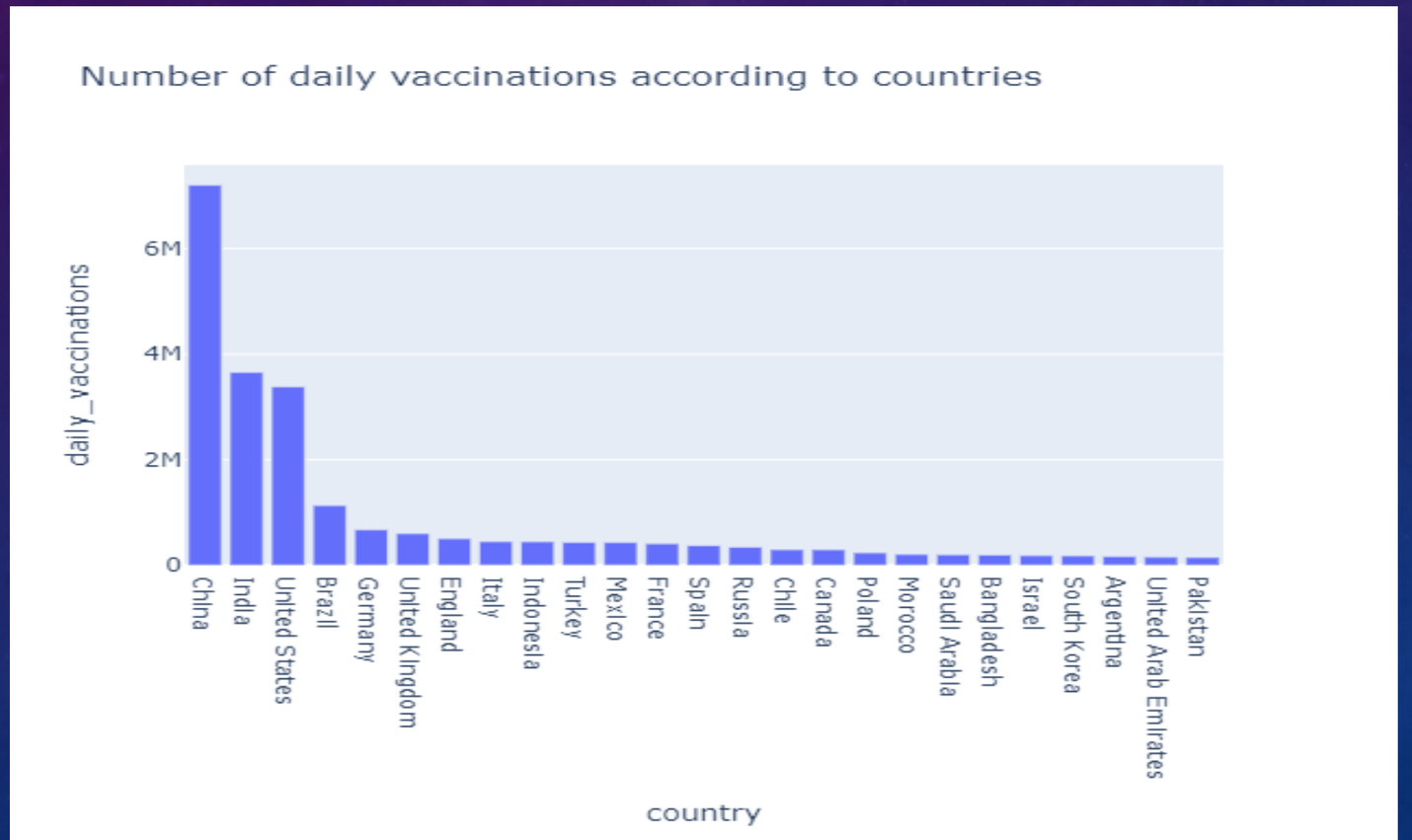

Total vs Total vaccinations per hundred grouped by Vaccines
total_vaccinations_per_hundred
200
150
100
50
0
0
100M
200M
300M
total_vaccinations

- **What is the number of total vaccinations & daily vaccinations according to countries?**

```
data = new_df[['country','total_vaccinations']].nlargest(25,'total_vaccinations') fig = px.bar(data, x = 'country',y = 'total_vaccinations',title="Number of total vaccinations according to countries",) fig.show( )
```


Number of total vaccinations according to countries
total_vaccinations
300M
250M
200M
150M
100M
50M
0
China
United States
India
United Kingdom
Brazil
England
Germany
Turkey
France
Italy
Indonesia
Russia
Mexico
Spain
Chile
Canada
Poland
United Arab Emirates
Israel
Saudi Arabia
Morocco
Bangladesh
Argentina
Hungary
Netherlands
country

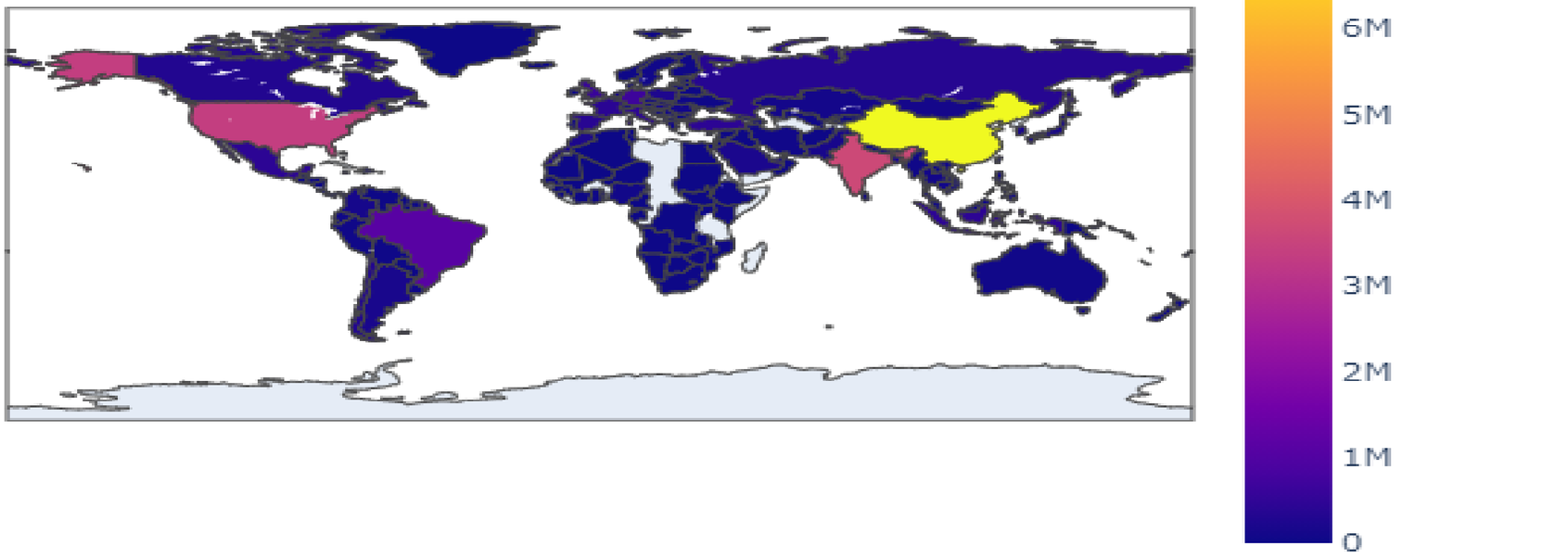
- *data = new_df[['country','daily_vaccinations']].nlargest(25,'daily_vaccinations') fig = px.bar(data, x = 'country',y = 'daily_vaccinations',title="Number of daily vaccinations according to countries",) fig.show()*



- ***Daily Vaccinations per Countries:***

```
fig = go.Choropleth( locations = new_df["country"],locationmode = 'country names', z = new_df['daily_vaccinations'], text= new_df['country'],colorbar = dict(title= "Daily Vaccinations")) data = [fig] layout = go.Layout(title = 'Daily Vaccinations per Countries') fig = dict(data = data,layout = layout) iplot(fig)
```

Daily Vaccinations per Countries



- ***The conclusions from the available data up to that point indicated that COVID-19 vaccines:***

  1.Are effective in preventing COVID-19 infection.

  2.Significantly reduce the risk of severe illness, hospitalization, and death.

- *3.Contribute to herd immunity, slowing the spread of the virus.*

- *2.Have demonstrated safety profiles, with side effects typically being mild and temporary.*

- *3.May require booster shots for ongoing protection due to the emergence of new variants.*