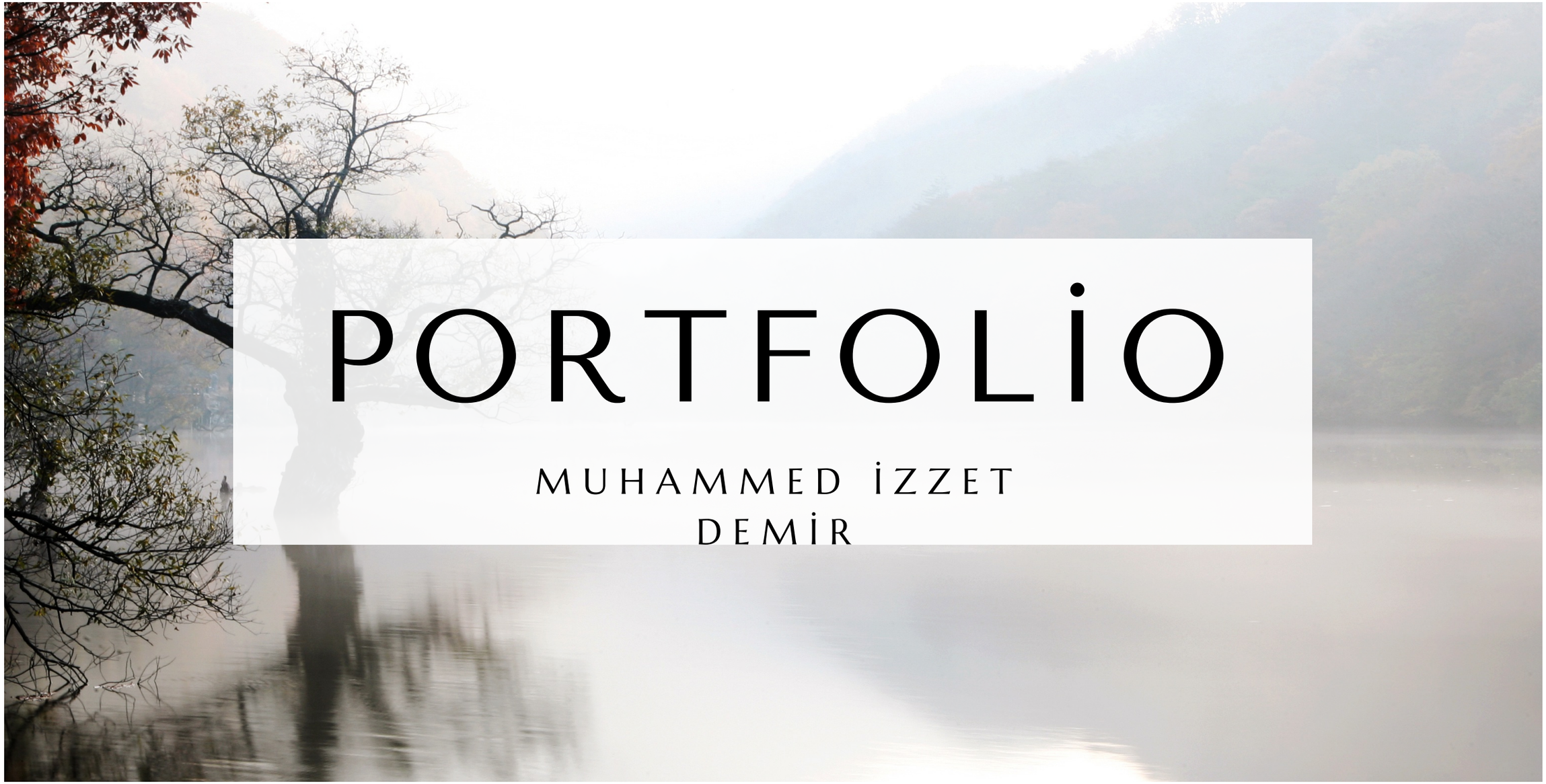
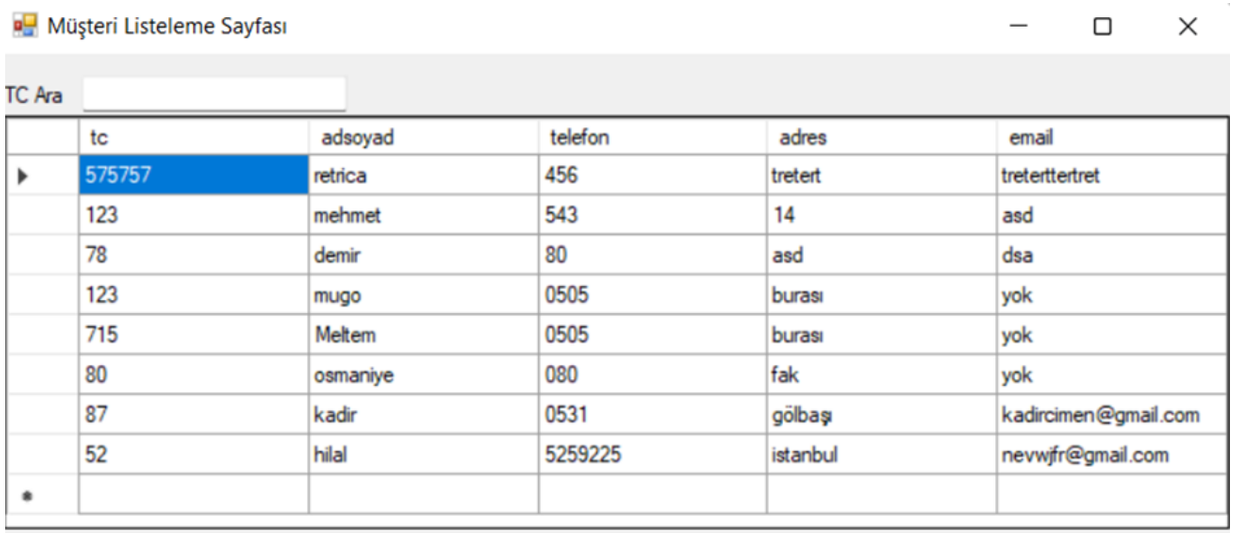


PORTFOLİO

MUHAMMED İZZET
DEMİR

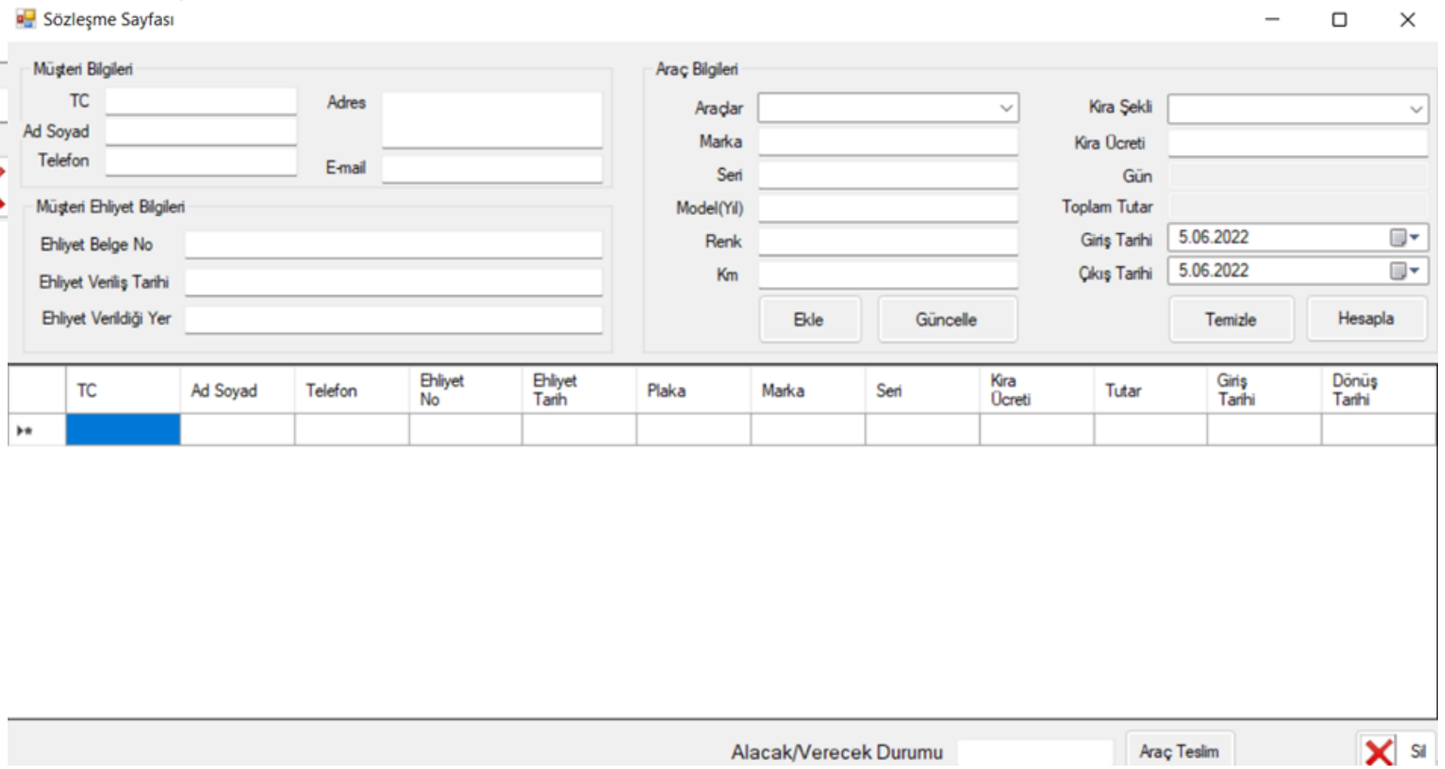
ABOUT ME

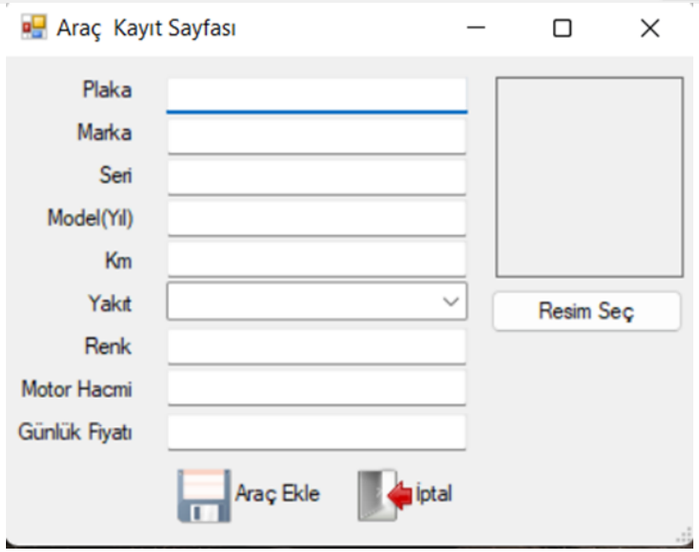
Gazi Üniversitesi Bilgisayar Mühendisliği 3.Sınıf
muhammedizzetdemirr@gmail.com
5052722937



PROJECT 01.

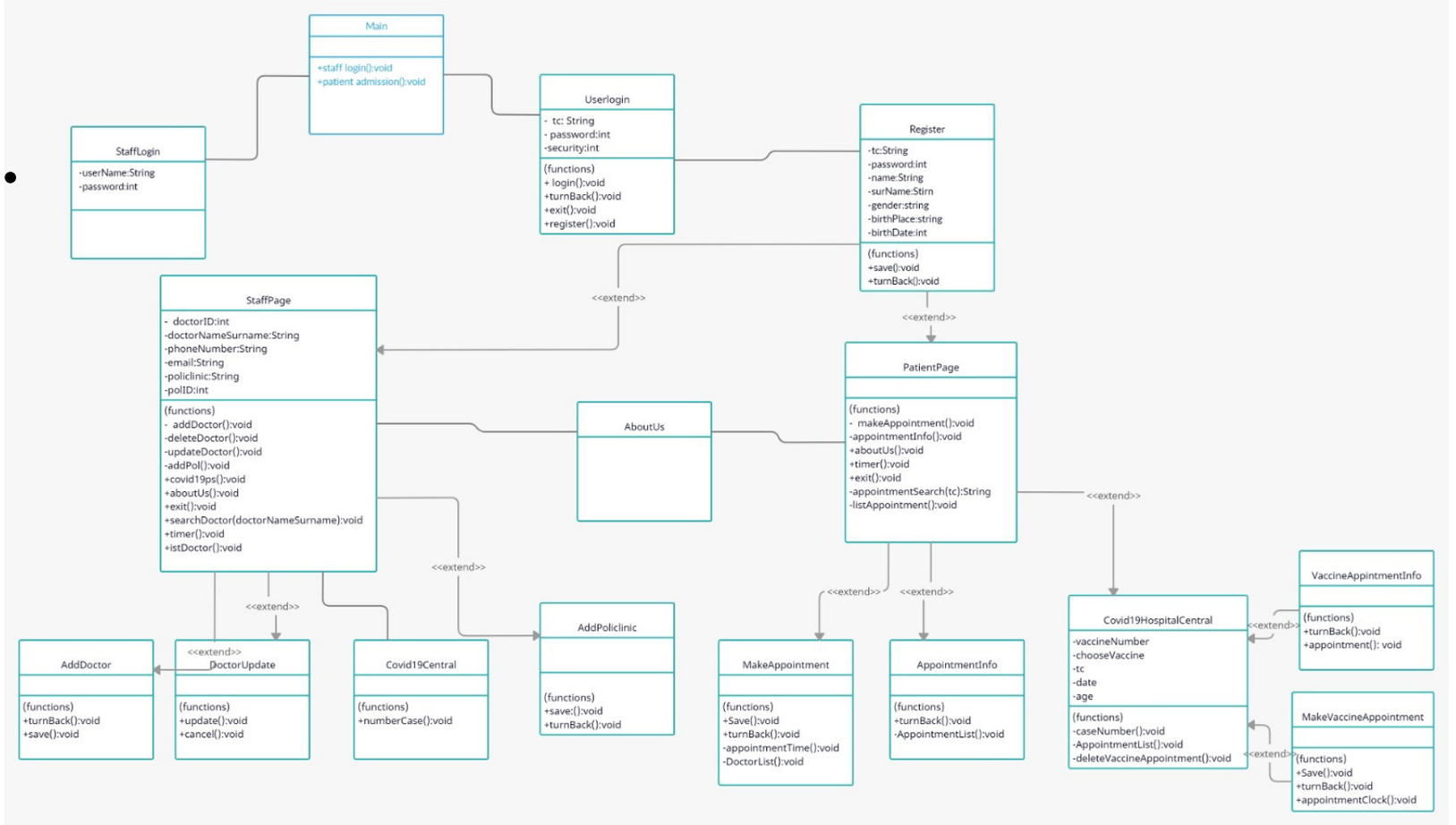
Araç Kiralama Otomasyonu
C# OOP





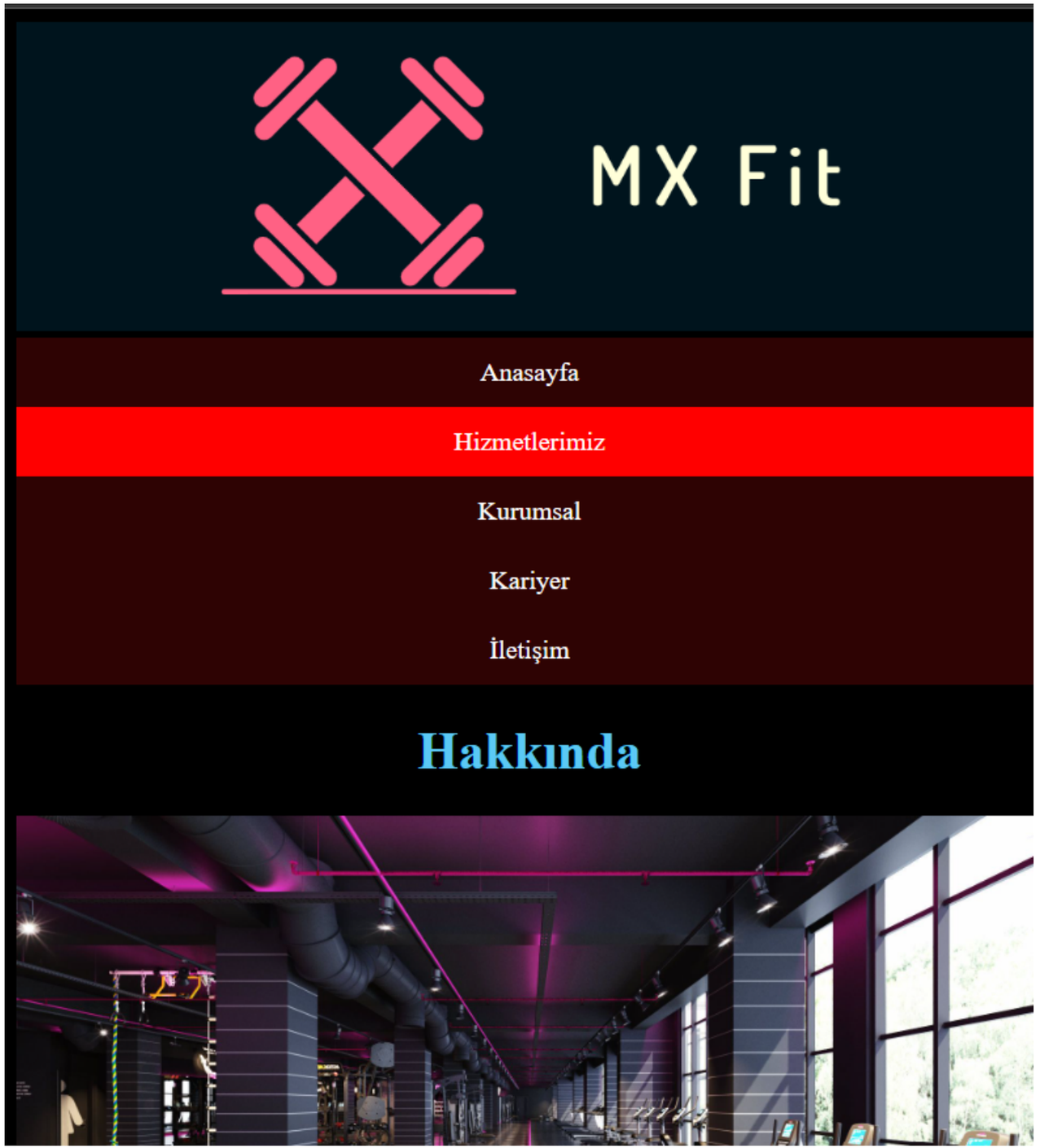
PROJECT 02.

HOSPİTAL MANGMENT SYSTEM JAVA



PROJECT 03.

Mx -fit Web sitesi (Only Use Html-Css)



PROJECT 04. Climate Change Machine Learning

ABSTRACT

In this paper, we describe our project topic, methodology, datasets, experiments, results, conclusions. In this project, we purpose to see climate change how effect on the natural environment. Therefore, we analyze about effects of climate change to make deduction about global temperature, gas emission, sea level and bird species.

Keywords

Climate change; Gas emissions; Sea level; Temperature; Bird species;

1. INTRODUCTION

Climate change, one of the biggest problems of humanity. It is one of the most important issues of the century we live in. Our primary motivation is to take care of our future by understanding this change and its effects, which will shape most of our lives. Another source of motivation is to raise people's awareness by explaining the consequences of climate change, for example, sea level, temperature and CO2 emissions. We aim to show the analysis about that relationship between temperature change and climate change and change most over the years since temperature changes.

2. METHODOLOGY

2.1 Temperature

We used px.choropleth to visualize earth temperatures on the map. "px.choropleth" is a Plotly Express function for creating choropleth maps. A choropleth map is a type of map that displays divided regions that are coloured or patterned in relation to a data given. The px.choropleth function makes it easy to create choropleth maps by providing options for customizing the map colors, labels, and markers.

We used plotly.graph_objs to visualize the average land temperature in Turkey and in World as a line graph. The "plotly.graph_objs" library is a module in the Plotly library that provides classes for creating and customizing various types of plots and charts using Plotly. It allows you to create a wide range of static, interactive, and animated visualizations using Python.

We used the LinearRegression() and polyfit() functions to make regression models. We also tested the accuracy of our model with the r^2_score() function.

2.2 CO2 Emission

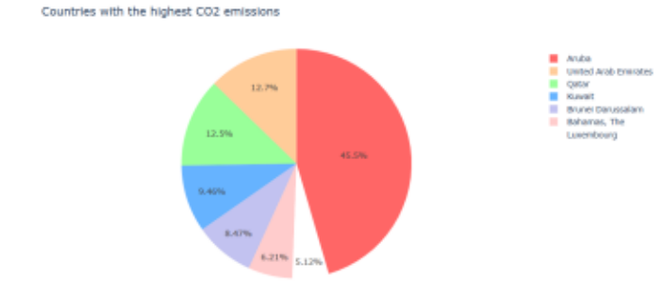
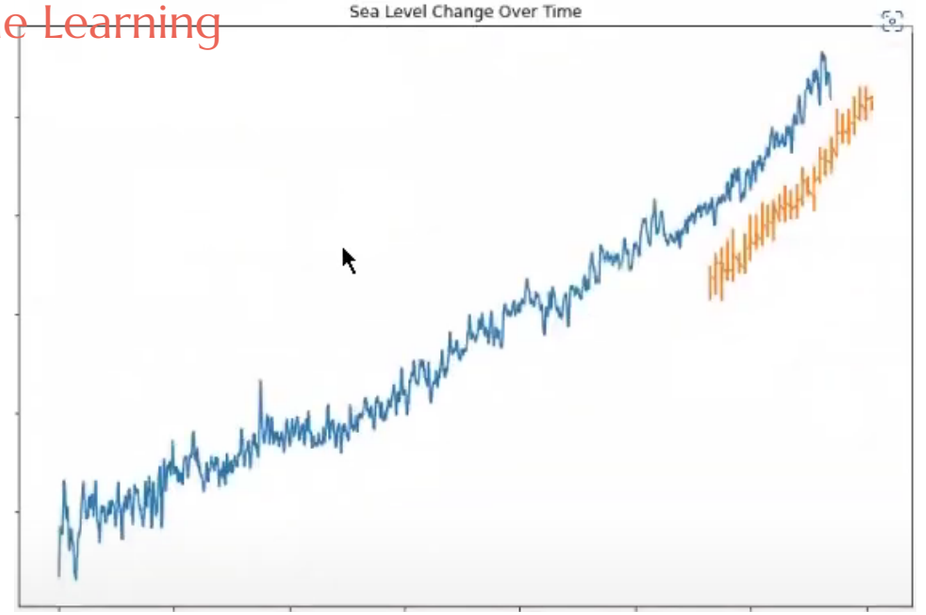
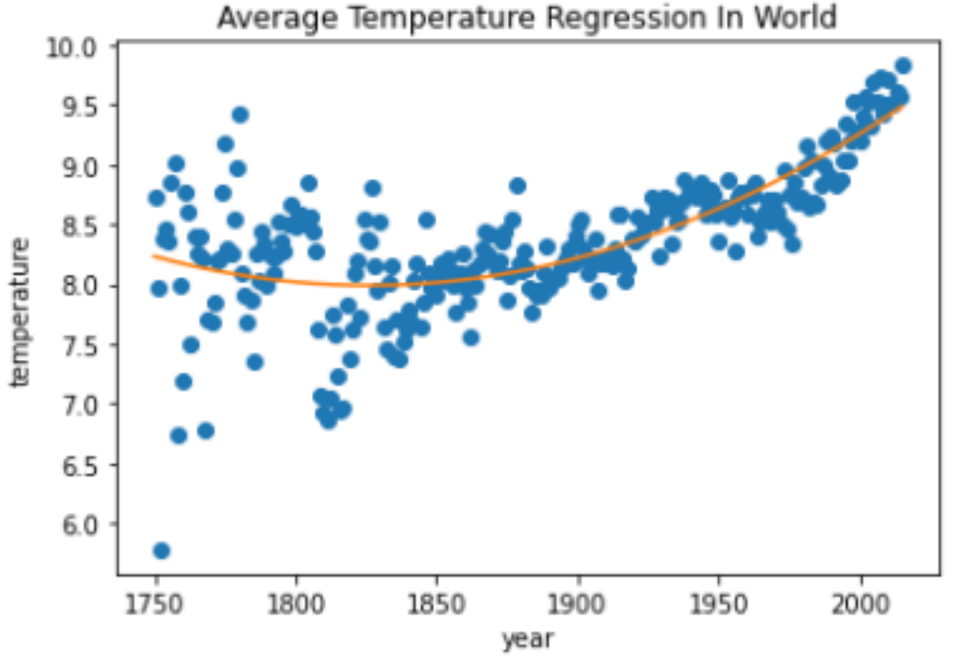
To summarize the data in the DataFrame, you can use the describe() method, which calculates various statistical measures

Trinidad and Tobago	Tunisia	Turkey	Tanzania	Uganda	Upper middle income	Uruguay	United States	St. Vincent and the Grenadines	Venezuela, RB	Vietnam
3.044399	0.413370	0.612271	0.082444	0.062317	2.573291	1.701585	15.999779	0.135865	7.009414	0.181947
5.329474	0.417045	0.616879	0.068047	0.058283	2.408432	1.602728	15.681256	0.133884	6.153191	0.183099
8.231023	0.417336	0.750243	0.071950	0.059458	2.370116	1.540660	16.013937	0.132162	6.188716	0.217694
1.467708	0.444553	0.767838	0.073218	0.057991	2.435563	1.639287	16.452762	0.174204	6.208593	0.196997
4.204809	0.618138	0.878790	0.091548	0.063657	2.523331	1.710104	16.968119	0.215409	6.041541	0.209870
5.699133	0.541833	0.884281	0.099910	0.078519	2.630170	2.049518	17.451725	0.170540	6.271781	0.217934
4.612122	0.621834	0.994631	0.114427	0.088752	2.741215	1.985095	18.121073	0.210973	5.690063	0.225434

for each numeric column as you can see the figure..

You can then use Seaborn to create a line plot showing the relationship between two continuous variables as a line. To do this, you can use the lineplot function from the Seaborn library.

```
top_countries = df2.sort_values(by='co2emission', ascending=False).head(150)
top_countries = top_countries.drop_duplicates('Country Name')
# Create the pie chart
fig = px.pie(top_countries, values='co2emission', names='Country Name', title='Countries with the highest CO2 emissions',
             color_discrete_sequence=['#ff6666', '#ffcc99', '#99ff99', '#66b3ff', '#c2c2f0'])
fig.show()
```



as you can see To create a pie chart visualizing the top countries with the highest and lowest CO2 emissions, you can sort the data in descending order based on the 'co2emission' column, select the top and bottom 9 countries, and use the pie chart function from a library like Matplotlib to create the chart.