

# ASSIGNMENT-1

```

    "semantic_type": "\",\n      "description": \"\"\n    }\n  },\n    {"\n      "column": "travel_time",\n      "properties": {\n        "dtype": "string",\n        "num_unique_values": 4,\n        "samples": [\n          "min",\n          "30-60 min",\n          "60-90 min",\n          ">90 min"],\n        "semantic_type": "\",\n      "description": \"\"\n    }\n  },\n    {"\n      "column": "extra_activities",\n      "properties": {\n        "dtype": "category",\n        "num_unique_values": 2,\n        "samples": [\n          "yes",\n          "no"],\n        "semantic_type": "\",\n      "description": \"\"\n    }\n  },\n    {"\n      "column": "study_method",\n      "properties": {\n        "dtype": "string",\n        "num_unique_values": 3,\n        "samples": [\n          "textbook",\n          "online",\n          "paper"],\n        "semantic_type": "\",\n      "description": \"\"\n    }\n  },\n    {"\n      "column": "math_score",\n      "properties": {\n        "dtype": "number",\n        "min": 27.457913249189204,\n        "max": 84.8,\n        "num_unique_values": 5,\n        "samples": [\n          57.6,\n          57.6,\n          57.6,\n          57.6,\n          57.6],\n        "semantic_type": "\",\n      "description": \"\"\n    }\n  },\n    {"\n      "column": "science_score",\n      "properties": {\n        "dtype": "number",\n        "std": 27.569312650118793,\n        "min": 27.5,\n        "max": 95.0,\n        "num_unique_values": 5,\n        "samples": [\n          68.8,\n          68.8,\n          68.8,\n          68.8,\n          68.8],\n        "semantic_type": "\",\n      "description": \"\"\n    }\n  },\n    {"\n      "column": "english_score",\n      "properties": {\n        "dtype": "number",\n        "std": 17.943466777632462,\n        "min": 30.0,\n        "max": 79.2,\n        "num_unique_values": 5,\n        "samples": [\n          64.8,\n          64.8,\n          64.8,\n          64.8,\n          64.8],\n        "semantic_type": "\",\n      "description": \"\"\n    }\n  },\n    {"\n      "column": "overall_score",\n      "properties": {\n        "dtype": "number",\n        "std": 23.94253537117571,\n        "min": 25.4,\n        "max": 89.6,\n        "num_unique_values": 5,\n        "samples": [\n          25.4,\n          25.4,\n          25.4,\n          25.4,\n          25.4],\n        "semantic_type": "\",\n      "description": \"\"\n    }\n  },\n    {"\n      "column": "final_grade",\n      "properties": {\n        "dtype": "string",\n        "num_unique_values": 4,\n        "samples": [\n          "A",\n          "B",\n          "C",\n          "D"],\n        "semantic_type": "\",\n      "description": \"\"\n    }\n  }\n},\n  "type": "dataframe"}\n\ndisplay(df.tail())
\n\n{"summary":{\n  "name": "display(df",\n  "rows": 5,\n  "fields": [\n    {"\n      "column": "student_id",\n      "properties": {\n        "dtype": "number",\n        "min": 4694,\n        "max": 12047,\n        "num_unique_values": 5,\n        "samples": [\n          1102,\n          1102,\n          1102,\n          1102,\n          1102]\n      }\n    }\n  ]\n},\n  "type": "dataframe"}\n
```



```

    "description": """\n        }\n    },\n    {\n        \"column\":\n        \"math_score\",\\n        \"properties\": {\n            \"dtype\":\n            \"number\",\\n            \"std\": 20.19472703455038,\\n            \"min\":\n            13.0,\\n            \"max\": 64.8,\\n            \"num_unique_values\": 5,\\n            \"samples\": [\n                64.8\n            ],\\n            \"semantic_type\":\n            \"\",\\n            \"description\": \"\"\n        }\n    },\n    {\n        \"column\": \"science_score\",\\n        \"properties\": {\n            \"dtype\": \"number\",\\n            \"std\": 12.088134678270258,\\n            \"min\": 20.3,\\n            \"max\": 48.5,\\n            \"num_unique_values\": 5,\\n            \"samples\": [\n                48.2\n            ],\\n            \"semantic_type\": \"\",\\n            \"description\": \"\"\n        }\n    },\n    {\n        \"column\": \"english_score\",\\n        \"properties\": {\n            \"dtype\": \"number\",\\n            \"std\": 19.333183907468527,\\n            \"min\": 7.3,\\n            \"max\": 52.3,\\n            \"num_unique_values\": 5,\\n            \"samples\": [\n                52.3\n            ],\\n            \"semantic_type\": \"\",\\n            \"description\": \"\"\n        }\n    },\n    {\n        \"column\": \"overall_score\",\\n        \"properties\": {\n            \"dtype\": \"number\",\\n            \"std\": 10.301844495040683,\\n            \"min\": 31.4,\\n            \"max\": 56.5,\\n            \"num_unique_values\": 5,\\n            \"samples\": [\n                56.5\n            ],\\n            \"semantic_type\": \"\",\\n            \"description\": \"\"\n        }\n    },\n    {\n        \"column\": \"final_grade\",\\n        \"properties\": {\n            \"dtype\": \"string\",\\n            \"num_unique_values\": 3,\\n            \"samples\": [\n                \"e\"\n            ],\\n            \"semantic_type\": \"\",\\n            \"description\": \"\"\n        }\n    }\n}\n},\"type\":\"dataframe\"}

print('DataFrame Shape:', df.shape)
print('\nDataFrame Columns:', df.columns.tolist())
print('\nDataFrame Info:')
df.info()

```

DataFrame Shape: (25000, 16)

DataFrame Columns: ['student\_id', 'age', 'gender', 'school\_type',  
'parent\_education', 'study\_hours', 'attendance\_percentage',  
'internet\_access', 'travel\_time', 'extra\_activities', 'study\_method',  
'math\_score', 'science\_score', 'english\_score', 'overall\_score',  
'final\_grade']

DataFrame Info:

#	Column	Non-Null Count	Dtype	
0	student_id	25000	non-null	int64
1	age	25000	non-null	int64
2	gender	25000	non-null	object
3	school_type	25000	non-null	object

```

4  parent_education      25000 non-null  object
5  study_hours           25000 non-null  float64
6  attendance_percentage 25000 non-null  float64
7  internet_access       25000 non-null  object
8  travel_time            25000 non-null  object
9  extra_activities        25000 non-null  object
10 study_method           25000 non-null  object
11 math_score              25000 non-null  float64
12 science_score           25000 non-null  float64
13 english_score            25000 non-null  float64
14 overall_score           25000 non-null  float64
15 final_grade             25000 non-null  object
dtypes: float64(6), int64(2), object(8)
memory usage: 3.1+ MB

display(df.describe(include='all'))

{"summary": {"\n    \"name\": \"display(df\", \n    \"rows\": 11,\n    \"fields\": [\n        {\n            \"column\": \"student_id\", \n            \"properties\": {\n                \"dtype\": \"number\", \n                \"std\": 7851.676509482255,\n                \"min\": 1.0,\n                \"max\": 25000.0,\n                \"num_unique_values\": 8,\n                \"samples\": [\n                    7493.0438,\n                    7461.5,\n                    25000.0\n                ],\n                \"semantic_type\": \"\", \n                \"description\": \"\"\n            }\n        },\n        {\n            \"column\": \"age\", \n            \"properties\": {\n                \"dtype\": \"number\", \n                \"std\": 8833.77621339813,\n                \"min\": 1.7038952176575877,\n                \"max\": 25000.0,\n                \"num_unique_values\": 8,\n                \"samples\": [\n                    16.48276,\n                    16.0,\n                    25000.0\n                ],\n                \"semantic_type\": \"\", \n                \"description\": \"\"\n            }\n        },\n        {\n            \"column\": \"gender\", \n            \"properties\": {\n                \"dtype\": \"category\", \n                \"num_unique_values\": 4,\n                \"samples\": [\n                    3,\n                    8463,\n                    \"25000\"\n                ],\n                \"semantic_type\": \"\", \n                \"description\": \"\"\n            }\n        },\n        {\n            \"column\": \"school_type\", \n            \"properties\": {\n                \"dtype\": \"category\", \n                \"num_unique_values\": 4,\n                \"samples\": [\n                    2,\n                    12725,\n                    25000\n                ],\n                \"semantic_type\": \"\", \n                \"description\": \"\"\n            }\n        },\n        {\n            \"column\": \"parent_education\", \n            \"properties\": {\n                \"dtype\": \"category\", \n                \"num_unique_values\": 4,\n                \"samples\": [\n                    6,\n                    4314,\n                    \"25000\"\n                ],\n                \"semantic_type\": \"\", \n                \"description\": \"\"\n            }\n        },\n        {\n            \"column\": \"study_hours\", \n            \"properties\": {\n                \"dtype\": \"number\", \n                \"std\": 8837.43496771498,\n                \"min\": 0.5,\n                \"max\": 25000.0,\n                \"num_unique_values\": 8,\n                \"samples\": [\n                    4.3,\n                    25000.0\n                ],\n                \"semantic_type\": \"\", \n                \"description\": \"\"\n            }\n        }\n    ]\n}
```

```

    },\n      {"column": "attendance_percentage",\n      "properties": {\n          "dtype": "number",\n          "min": 14.373170735868388,\n          "max": 25000.0,\n          "samples": [\n              8815.394124016382,\n              25000.0\n          ],\n          "semantic_type": "\\",\\n\n      },\n      {"column": "internet_access",\n      "properties": {\n          "dtype": "category",\n          "num_unique_values": 4,\n          "samples": [\n              2,\n              21227,\n              75.084084,\n              75.1,\n              25000\n          ],\n          "semantic_type": "\\",\\n\n      },\n      {"column": "travel_time",\n      "properties": {\n          "dtype": "category",\n          "num_unique_values": 4,\n          "samples": [\n              4,\n              6362,\n              25000,\n              25000\n          ],\n          "semantic_type": "\\",\\n\n      },\n      {"column": "extra_activities",\n      "properties": {\n          "dtype": "category",\n          "num_unique_values": 4,\n          "samples": [\n              2,\n              12500,\n              25000,\n              25000\n          ],\n          "semantic_type": "\\",\\n\n      },\n      {"column": "study_method",\n      "properties": {\n          "dtype": "category",\n          "num_unique_values": 4,\n          "samples": [\n              6,\n              4341,\n              25000,\n              25000\n          ],\n          "semantic_type": "\\",\\n\n      },\n      {"column": "math_score",\n      "properties": {\n          "dtype": "number",\n          "min": 0.0,\n          "max": 25000.0,\n          "samples": [\n              63.78594399999999,\n              64.1,\n              25000.0\n          ],\n          "semantic_type": "\\",\\n\n      },\n      {"column": "science_score",\n      "properties": {\n          "dtype": "number",\n          "min": 0.0,\n          "max": 25000.0,\n          "samples": [\n              8819.849517683746,\n              64.1,\n              25000.0\n          ],\n          "semantic_type": "\\",\\n\n      },\n      {"column": "english_score",\n      "properties": {\n          "dtype": "number",\n          "min": 0.0,\n          "max": 25000.0,\n          "samples": [\n              63.74532000000001,\n              64.1,\n              25000.0\n          ],\n          "semantic_type": "\\",\\n\n      },\n      {"column": "overall_score",\n      "properties": {\n          "dtype": "number",\n          "min": 14.5,\n          "max": 25000.0,\n          "samples": [\n              63.68194800000006,\n              64.2,\n              25000.0\n          ],\n          "semantic_type": "\\",\\n\n      },\n      {"column": "num_unique_values",\n      "properties": {\n          "dtype": "number",\n          "min": 8,\n          "max": 25000.0,\n          "samples": [\n              64.006172,\n              64.2,\n              25000.0\n          ],\n          "semantic_type": "\\",\\n\n      }\n  ]\n}\n]

```

```

  "semantic_type": "\",\n      "description": \"\"\n    }\n  },\n  {\n    "column": "final_grade",\n    "properties": {\n      "dtype": "category",\n      "num_unique_values": 4,\n      "samples": [\n        6,\n        6311,\n        25000\n      ],\n      "semantic_type": "\","description": \"\"\n    }\n  }\n}\n},\n"type": "dataframe"

```

## ASSIGNMENT-2

```

score_columns = [col for col in df.columns if 'score' in col]
scores_df = df[score_columns]
display(scores_df.head())

{"summary":{\n  "name": "display(scores_df)",\n  "rows": 5,\n  "fields": [\n    {"column": "math_score",\n      "properties": {\n        "dtype": "number",\n        "std": 27.457913249189204,\n        "min": 8.9,\n        "max": 84.8,\n        "num_unique_values": 5,\n        "samples": [\n          57.6,\n          8.9,\n          84.8\n        ],\n        "semantic_type": "\","description": \"\"\n      }\n    },\n    {"column": "science_score",\n      "properties": {\n        "dtype": "number",\n        "std": 27.569312650118793,\n        "min": 27.5,\n        "max": 95.0,\n        "num_unique_values": 5,\n        "samples": [\n          68.8,\n          32.7,\n          95.0\n        ],\n        "semantic_type": "\","description": \"\"\n      }\n    },\n    {"column": "english_score",\n      "properties": {\n        "dtype": "number",\n        "std": 17.943466777632462,\n        "min": 30.0,\n        "max": 79.2,\n        "num_unique_values": 5,\n        "samples": [\n          64.8,\n          30.0,\n          79.2\n        ],\n        "semantic_type": "\","description": \"\"\n      }\n    },\n    {"column": "overall_score",\n      "properties": {\n        "dtype": "number",\n        "std": 23.94253537117571,\n        "min": 25.4,\n        "max": 89.6,\n        "num_unique_values": 5,\n        "samples": [\n          89.6,\n          61.3,\n          25.4\n        ],\n        "semantic_type": "\","description": \"\"\n      }\n    }\n  ]\n},\n"type": "dataframe"}\n<google.colab._quickchart_helpers.SectionTitle at 0x79ac379ac980>\n\nfrom matplotlib import pyplot as plt\n_df_0['math_score'].plot(kind='hist', bins=20, title='math_score')\nplt.gca().spines[['top', 'right']].set_visible(False)\n\nfrom matplotlib import pyplot as plt\n_df_1['science_score'].plot(kind='hist', bins=20,\ntitle='science_score')\nplt.gca().spines[['top', 'right']].set_visible(False)

```

```
from matplotlib import pyplot as plt
_df_2['english_score'].plot(kind='hist', bins=20,
title='english_score')
plt.gca().spines[['top', 'right']].set_visible(False)

from matplotlib import pyplot as plt
_df_3['overall_score'].plot(kind='hist', bins=20,
title='overall_score')
plt.gca().spines[['top', 'right']].set_visible(False)

<google.colab._quickchart_helpers.SectionTitle at 0x79ac40232510>

from matplotlib import pyplot as plt
_df_4.plot(kind='scatter', x='math_score', y='science_score', s=32,
alpha=.8)
plt.gca().spines[['top', 'right']].set_visible(False)

from matplotlib import pyplot as plt
_df_5.plot(kind='scatter', x='science_score', y='english_score', s=32,
alpha=.8)
plt.gca().spines[['top', 'right']].set_visible(False)

from matplotlib import pyplot as plt
_df_6.plot(kind='scatter', x='english_score', y='overall_score', s=32,
alpha=.8)
plt.gca().spines[['top', 'right']].set_visible(False)

<google.colab._quickchart_helpers.SectionTitle at 0x79ac33df07a0>

from matplotlib import pyplot as plt
_df_7['math_score'].plot(kind='line', figsize=(8, 4),
title='math_score')
plt.gca().spines[['top', 'right']].set_visible(False)

from matplotlib import pyplot as plt
_df_8['science_score'].plot(kind='line', figsize=(8, 4),
title='science_score')
plt.gca().spines[['top', 'right']].set_visible(False)

from matplotlib import pyplot as plt
_df_9['english_score'].plot(kind='line', figsize=(8, 4),
title='english_score')
plt.gca().spines[['top', 'right']].set_visible(False)

from matplotlib import pyplot as plt
_df_10['overall_score'].plot(kind='line', figsize=(8, 4),
title='overall_score')
plt.gca().spines[['top', 'right']].set_visible(False)

high_math_scores_df = df[df['math_score'] > 70]
print(f"Number of students with math score above 70:
```

```

{len(high_math_scores_df)})")
display(high_math_scores_df.head())

Number of students with math score above 70: 10108

{
  "summary": {
    "name": "display(high_math_scores_df)",
    "rows": 5,
    "fields": [
      {
        "column": "student_id",
        "properties": {
          "dtype": "number",
          "std": 4,
          "min": 3,
          "max": 15,
          "num_unique_values": 5,
          "samples": [10, 15, 11]
        },
        "semantic_type": "\",
        "description": "\n"
      },
      {
        "column": "age",
        "properties": {
          "dtype": "number",
          "std": 1,
          "min": 14,
          "max": 18,
          "num_unique_values": 3,
          "samples": [17, 14, 18]
        },
        "semantic_type": "\",
        "description": "\n"
      },
      {
        "column": "gender",
        "properties": {
          "dtype": "category",
          "num_unique_values": 2,
          "samples": ["other", "female"]
        },
        "semantic_type": "\",
        "description": "\n"
      },
      {
        "column": "school_type",
        "properties": {
          "dtype": "category",
          "num_unique_values": 2,
          "samples": ["public", "private"]
        },
        "semantic_type": "\",
        "description": "\n"
      },
      {
        "column": "parent_education",
        "properties": {
          "dtype": "string",
          "num_unique_values": 4,
          "samples": ["high school", ""]
        },
        "semantic_type": "\",
        "description": "\n"
      },
      {
        "column": "study_hours",
        "properties": {
          "dtype": "number",
          "std": 1.1022703842524302,
          "min": 4.9,
          "max": 7.9,
          "num_unique_values": 4,
          "samples": [6.8, 4.9]
        },
        "semantic_type": "\",
        "description": "\n"
      },
      {
        "column": "attendance_percentage",
        "properties": {
          "dtype": "number",
          "std": 14.519538560160925,
          "min": 58.2,
          "max": 90.5,
          "num_unique_values": 5,
          "samples": [85.3, 62.4]
        },
        "semantic_type": "\",
        "description": "\n"
      },
      {
        "column": "internet_access",
        "properties": {
          "dtype": "category",
          "num_unique_values": 2,
          "samples": ["no", "yes"]
        },
        "semantic_type": "\",
        "description": "\n"
      },
      {
        "column": "travel_time",
        "properties": {
          "dtype": "string",
          "num_unique_values": 3,
          "samples": ["<15"]
        },
        "semantic_type": "\",
        "description": "\n"
      }
    ]
  }
}

```

```

min\", \n      \">60 min\" \n      ], \n      \"semantic_type\": \n      \"\",
\"\", \n      \"description\": \"\\n      }, \n      {\n      \"column\": \"extra_activities\", \n      \"properties\": {\n      \"dtype\": \"category\", \n      \"num_unique_values\": 1, \n      \"samples\": [\n      \"no\", \n      ], \n      \"semantic_type\": \"\", \n      \"description\": \"\\n      } \n      }, \n      {\n      \"column\": \"study_method\", \n      \"properties\": {\n      \"dtype\": \"string\", \n      \"num_unique_values\": 3, \n      \"samples\": [\n      \"notes\", \n      ], \n      \"semantic_type\": \"\", \n      \"description\": \"\\n      } \n      }, \n      {\n      \"column\": \"math_score\", \n      \"properties\": {\n      \"dtype\": \"number\", \n      \"std\": 6.9823348530416345, \n      \"min\": 71.9, \n      \"max\": 88.1, \n      \"num_unique_values\": 5, \n      \"samples\": [\n      71.9\n      ], \n      \"semantic_type\": \"\", \n      \"description\": \"\\n      } \n      }, \n      {\n      \"column\": \"science_score\", \n      \"properties\": {\n      \"dtype\": \"number\", \n      \"std\": 11.316934213823107, \n      \"min\": 66.1, \n      \"max\": 95.0, \n      \"num_unique_values\": 5, \n      \"samples\": [\n      70.4\n      ], \n      \"semantic_type\": \"\", \n      \"description\": \"\\n      } \n      }, \n      {\n      \"column\": \"english_score\", \n      \"properties\": {\n      \"dtype\": \"number\", \n      \"std\": 9.891056566413923, \n      \"min\": 56.2, \n      \"max\": 81.3, \n      \"num_unique_values\": 5, \n      \"samples\": [\n      81.3\n      ], \n      \"semantic_type\": \"\", \n      \"description\": \"\\n      } \n      }, \n      {\n      \"column\": \"overall_score\", \n      \"properties\": {\n      \"dtype\": \"number\", \n      \"std\": 8.348472914252042, \n      \"min\": 69.3, \n      \"max\": 89.6, \n      \"num_unique_values\": 5, \n      \"samples\": [\n      69.6\n      ], \n      \"semantic_type\": \"\", \n      \"description\": \"\\n      } \n      }, \n      {\n      \"column\": \"final_grade\", \n      \"properties\": {\n      \"dtype\": \"string\", \n      \"num_unique_values\": 3, \n      \"samples\": [\n      \"b\"\n      ], \n      \"semantic_type\": \"\", \n      \"description\": \"\\n      } \n      } \n    ], \n    \"type\": \"dataframe\"}

female_students_df = df[df['gender'] == 'female']
print(f"Number of female students: {len(female_students_df)}")
display(female_students_df.head())

```

Number of female students: 8290

```

{ "summary": {
  "name": "display(female_students_df)", 
  "rows": 5, 
  "fields": [
    {
      "column": "student_id", 
      "properties": {
        "dtype": "number", 
        "std": 2, 
        "min": 2, 
        "max": 8, 
        "num_unique_values": 5, 
        "samples": [
          3, 
          8, 
          5
        ], 
        "semantic_type": "", 
        "description": ""
      }
    }
  ]
}

```

```

    },\n      {\n        \\"column\": \\"age\\",\n          \\"properties\": {\n            \\"dtype\\": \\"number\\",\n              \\"std\\": 1,\n              \\"min\\": 14,\n              \\"max\\": 18,\n              \\"num_unique_values\\": 4,\n              \\"samples\\": [\n                17,\n                14,\n                18\n              ],\n            \\"semantic_type\\": \\"\\",\n              \\"description\\": \\"\\n            \\"\n          }\n        },\n        {\n          \\"column\": \\"gender\\",\n            \\"properties\": {\n              \\"dtype\\": \\"category\\",\n                \\"num_unique_values\\": 1,\n                \\"samples\\": [\n                  \\"female\\"
                ],\n              \\"semantic_type\\": \\"\\",\n                \\"description\\": \\"\\n            \\"\n            \"},\n            \\"column\": \\"school_type\\",\n              \\"properties\": {\n                \\"dtype\\": \\"category\\",\n                  \\"num_unique_values\\": 2,\n                  \\"samples\\": [\n                    \\"private\\"
                  ],\n                  \\"semantic_type\\": \\"\\",\n                    \\"description\\": \\"\\n                \"},\n                    \\"column\": \\"parent_education\\",\n                      \\"properties\": {\n                        \\"dtype\\": \\"string\\",\n                          \\"num_unique_values\\": 3,\n                          \\"samples\\": [\n                            \\"graduate\\"
                            ],\n                            \\"semantic_type\\": \\"\\",\n                              \\"description\\": \\"\\n                \"},\n                              \\"column\": \\"study_hours\\",\n                                \\"properties\": {\n                                  \\"dtype\\": \\"number\\",\n                                    \\"std\\": 2.7373344698812385,\n                                    \\"min\\": 1.3,\n                                    \\"max\\": 7.9,\n                                    \\"num_unique_values\\": 5,\n                                    \\"samples\\": [\n                                      7.9
                                    ],\n                                    \\"semantic_type\\": \\"\\",\n                                      \\"description\\": \\"\\n                \"},\n                                      \\"column\": \\"attendance_percentage\\",\n                                        \\"properties\": {\n                                          \\"dtype\\": \\"number\\",\n                                            \\"std\\": 12.83931462345245,\n                                            \\"min\\": 59.4,\n                                            \\"max\\": 87.8,\n                                            \\"num_unique_values\\": 5,\n                                            \\"samples\\": [\n                                              65.5
                                            ],\n                                            \\"semantic_type\\": \\"\\",\n                                              \\"description\\": \\"\\n                \"},\n                                              \\"column\": \\"internet_access\\",\n                                                \\"properties\": {\n                                                  \\"dtype\\": \\"category\\",\n                                                    \\"num_unique_values\\": 2,\n                                                    \\"samples\\": [\n                                                      \\"no\\"
                                                      ],\n                                                      \\"semantic_type\\": \\"\\",\n                                                        \\"description\\": \\"\\n                \"},\n                                                        \\"column\": \\"travel_time\\",\n                                                          \\"properties\": {\n                                                            \\"dtype\\": \\"string\\",\n                                                              \\"num_unique_values\\": 3,\n                                                              \\"samples\\": [\n                                                                \">>60
                                                                min
                                                                ],\n                                                                \\"semantic_type\\": \\"\\",\n                                                                  \\"description\\": \\"\\n                \"},\n                                                                  \\"column\": \\"extra_activities\\",\n                        \\"properties\": {\n                          \\"dtype\\": \\"category\\",\n                            \\"num_unique_values\\": 2,\n                            \\"samples\\": [\n                              \\"yes\\"
                              ],\n                              \\"semantic_type\\": \\"\\",\n                                \\"description\\": \\"\\n                \"},\n                                \\"column\": \\"study_method\\",\n                                  \\"properties\": {\n                                    \\"dtype\\": \\"string\\",\n                                      \\"num_unique_values\\": 3,\n                                      \\"samples\\": [\n                                        \\"textbook\\"
                                        ],\n                                        \\"semantic_type\\": \\"\\",\n                                          \\"description\\": \\"\\n                \"},\n                                          \\"column\": \\"math_score\\",\n                                            \\"properties\": {\n                                              \\"dtype\\": \\"number\\",\n                                                \\"std\\": 27.697057605456934,\n                                                \\"min\\": 8.9,\n                                                \\"max\\": 84.8,\n                                                \\"num_unique_values\\": 5,\n                                                \\"samples\\": [\n                                                  84.8
                                                ],\n                                                \\"semantic_type\\": \\"\\",\n                                                  \\"description\\": \\"\\n                \"}\n      }\n    }\n  }\n}\n
```

```
\"semantic_type\": \"\",\\n      \"description\": \"\"\n    },\\n    {\\n      \"column\": \"science_score\",\\n      \"properties\": {\\n        \"dtype\": \"number\",\\n        \"std\": 27.13346420640026,\\n        \"min\": 29.4,\\n        \"max\": 95.0,\\n        \"num_unique_values\": 5,\\n        \"samples\": [\n          95.0\n        ],\\n        \"semantic_type\": \"\",\\n        \"description\": \"\"\n      },\\n      {\\n        \"column\": \"english_score\",\\n        \"properties\": {\\n          \"dtype\": \"number\",\\n          \"std\": 19.651768368266506,\\n          \"min\": 30.0,\\n          \"max\": 79.2,\\n          \"num_unique_values\": 5,\\n          \"samples\": [\n            79.2\n          ],\\n          \"semantic_type\": \"\",\\n          \"description\": \"\"\n        },\\n        {\\n          \"column\": \"overall_score\",\\n          \"properties\": {\\n            \"dtype\": \"number\",\\n            \"std\": 25.293180899206803,\\n            \"min\": 25.4,\\n            \"max\": 89.6,\\n            \"num_unique_values\": 5,\\n            \"samples\": [\n              89.6\n            ],\\n            \"semantic_type\": \"\",\\n            \"description\": \"\"\n          },\\n          {\\n            \"column\": \"final_grade\",\\n            \"properties\": {\\n              \"dtype\": \"string\",\\n              \"num_unique_values\": 3,\\n              \"samples\": [\n                \"d\"\n              ],\\n              \"semantic_type\": \"\",\\n              \"description\": \"\"\n            }\n          }\n        ]\n      }\n    },\\n    \"type\": \"dataframe\"\n}\n\nprint("Count of students by gender:")
display(df['gender'].value_counts())
```

```
print("\nCount of students by school type:")
display(df['school_type'].value_counts())
```

```
print("\nCount of students by internet access:")
display(df['internet_access'].value_counts())
```

Count of students by gender:

```
gender
other     8463
female    8290
male      8247
Name: count, dtype: int64
```

Count of students by school type:

```
school_type
private   12725
public    12275
Name: count, dtype: int64
```

Count of students by internet access:

```
internet_access
yes    21227
no     3773
Name: count, dtype: int64
```

### ASSIGNMENT-3

```
import numpy as np
import pandas as pd

if 'scores_df' not in locals():
    score_columns = [col for col in df.columns if 'score' in col]
    scores_df = df[score_columns]

math_scores_np = scores_df['math_score'].to_numpy()
science_scores_np = scores_df['science_score'].to_numpy()
english_scores_np = scores_df['english_score'].to_numpy()
overall_scores_np = scores_df['overall_score'].to_numpy()

print("Score columns converted to NumPy arrays.")

Score columns converted to NumPy arrays.

print("--- Math Scores ---")
print(f"Mean Math Score: {np.mean(math_scores_np):.2f}")
print(f"Median Math Score: {np.median(math_scores_np):.2f}")
print(f"Standard Deviation Math Score: {np.std(math_scores_np):.2f}")

print("\n--- Science Scores ---")
print(f"Mean Science Score: {np.mean(science_scores_np):.2f}")
print(f"Median Science Score: {np.median(science_scores_np):.2f}")
print(f"Standard Deviation Science Score: {np.std(science_scores_np):.2f}")

print("\n--- English Scores ---")
print(f"Mean English Score: {np.mean(english_scores_np):.2f}")
print(f"Median English Score: {np.median(english_scores_np):.2f}")
print(f"Standard Deviation English Score: {np.std(english_scores_np):.2f}")

print("\n--- Overall Scores ---")
print(f"Mean Overall Score: {np.mean(overall_scores_np):.2f}")
print(f"Median Overall Score: {np.median(overall_scores_np):.2f}")
print(f"Standard Deviation Overall Score: {np.std(overall_scores_np):.2f}")

--- Math Scores ---
Mean Math Score: 63.79
Median Math Score: 64.10
Standard Deviation Math Score: 20.87
```

```
--- Science Scores ---
Mean Science Score: 63.75
Median Science Score: 64.10
Standard Deviation Science Score: 20.97

--- English Scores ---
Mean English Score: 63.68
Median English Score: 64.20
Standard Deviation English Score: 20.79

--- Overall Scores ---
Mean Overall Score: 64.01
Median Overall Score: 64.20
Standard Deviation Overall Score: 18.93

print("--- Math Scores ---")
print(f"Minimum Math Score: {np.min(math_scores_np):.2f}")
print(f"Maximum Math Score: {np.max(math_scores_np):.2f}")

print("\n--- Science Scores ---")
print(f"Minimum Science Score: {np.min(science_scores_np):.2f}")
print(f"Maximum Science Score: {np.max(science_scores_np):.2f}")

print("\n--- English Scores ---")
print(f"Minimum English Score: {np.min(english_scores_np):.2f}")
print(f"Maximum English Score: {np.max(english_scores_np):.2f}")

print("\n--- Overall Scores ---")
print(f"Minimum Overall Score: {np.min(overall_scores_np):.2f}")
print(f"Maximum Overall Score: {np.max(overall_scores_np):.2f}")

--- Math Scores ---
Minimum Math Score: 0.00
Maximum Math Score: 100.00

--- Science Scores ---
Minimum Science Score: 0.00
Maximum Science Score: 100.00

--- English Scores ---
Minimum English Score: 0.00
Maximum English Score: 100.00

--- Overall Scores ---
Minimum Overall Score: 14.50
Maximum Overall Score: 100.00
```

ASSIGNMENT-4

```
import matplotlib.pyplot as plt
import seaborn as sns

# Set the style for the plots
sns.set_style("whitegrid")

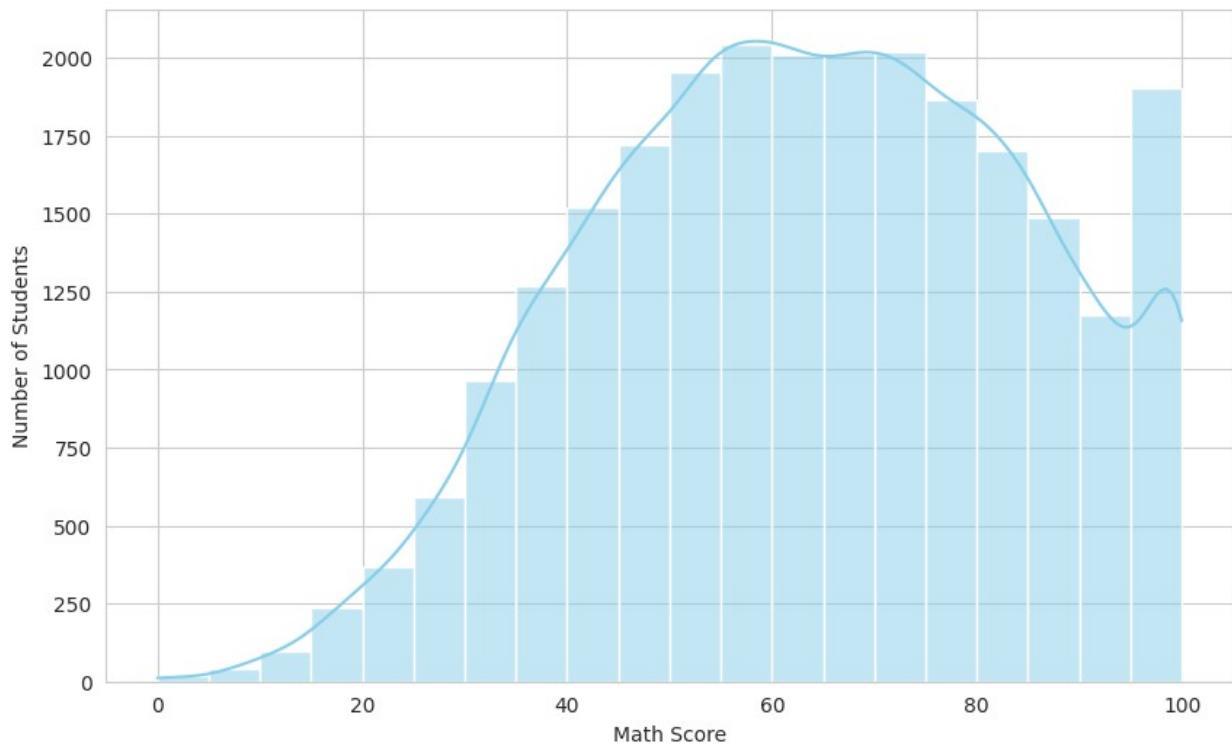
# Task 1: Plot histogram of math scores
plt.figure(figsize=(10, 6))
sns.histplot(df['math_score'], bins=20, kde=True, color='skyblue')
plt.title('Distribution of Math Scores')
plt.xlabel('Math Score')
plt.ylabel('Number of Students')
plt.show()

# Task 2: Plot bar chart of average scores
average_scores = df[['math_score', 'science_score', 'english_score',
'overall_score']].mean()

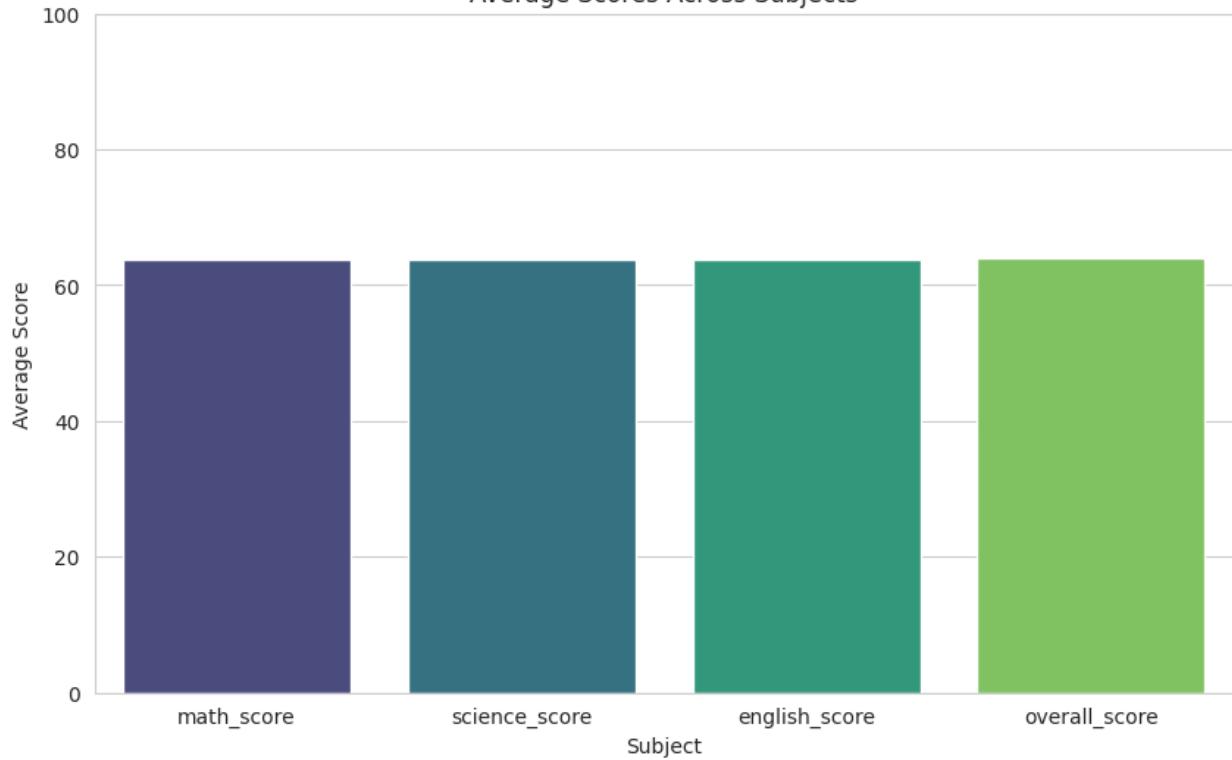
plt.figure(figsize=(10, 6))
sns.barplot(x=average_scores.index, y=average_scores.values,
hue=average_scores.index, palette='viridis', legend=False)
plt.title('Average Scores Across Subjects')
plt.xlabel('Subject')
plt.ylabel('Average Score')
plt.ylim(0, 100) # Scores are typically out of 100
plt.show()

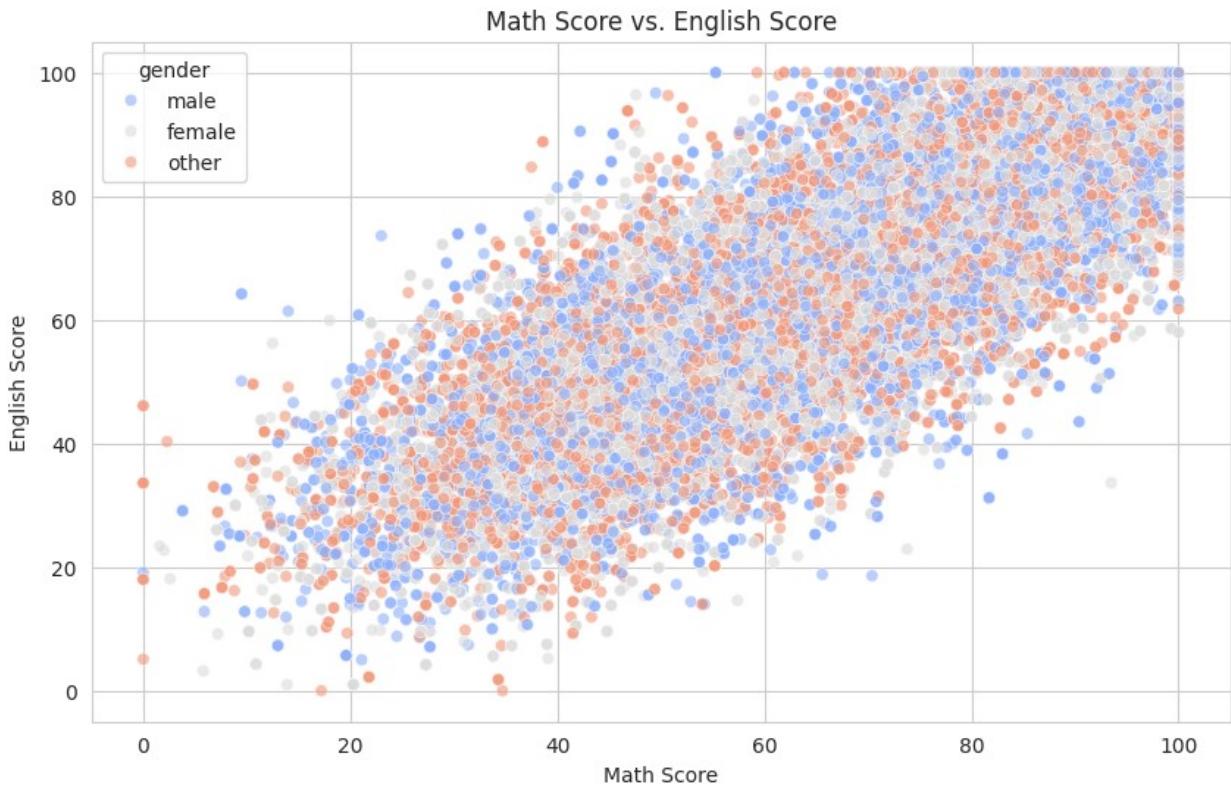
# Task 3: Create scatter plot between math and writing scores
# (assuming English score is writing score)
plt.figure(figsize=(10, 6))
sns.scatterplot(x=df['math_score'], y=df['english_score'],
hue=df['gender'], palette='coolwarm', alpha=0.6)
plt.title('Math Score vs. English Score')
plt.xlabel('Math Score')
plt.ylabel('English Score')
plt.show()
```

Distribution of Math Scores



Average Scores Across Subjects





## ASSIGNMENT-5

```

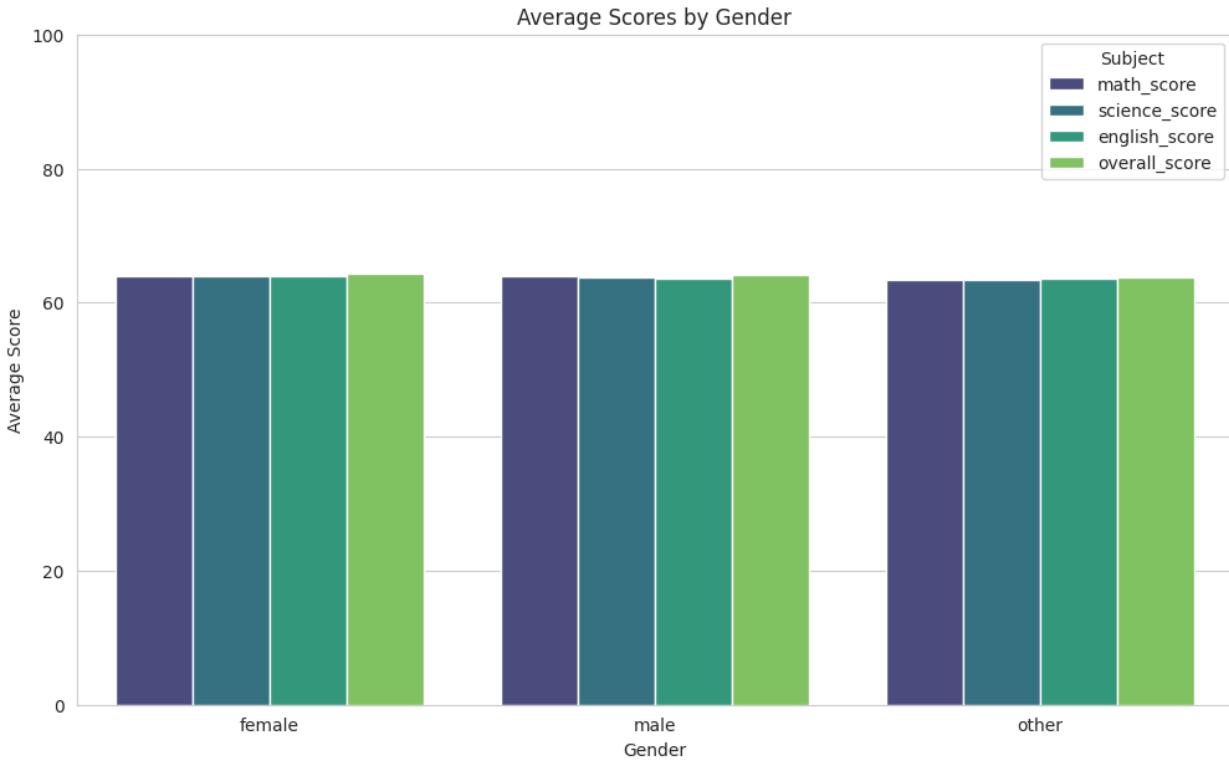
import matplotlib.pyplot as plt
import seaborn as sns

sns.set_style("whitegrid")

gender_scores = df.groupby('gender')[['math_score', 'science_score',
'english_score', 'overall_score']].mean().reset_index()

plt.figure(figsize=(12, 7))
gender_scores_melted = gender_scores.melt(id_vars='gender',
var_name='Subject', value_name='Average Score')
sns.barplot(x='gender', y='Average Score', hue='Subject',
data=gender_scores_melted, palette='viridis')
plt.title('Average Scores by Gender')
plt.xlabel('Gender')
plt.ylabel('Average Score')
plt.ylim(0, 100)
plt.legend(title='Subject')
plt.show()

```



```
parent_education_scores = df.groupby('parent_education')[['math_score', 'science_score', 'english_score', 'overall_score']].mean().reset_index()

plt.figure(figsize=(14, 8))
parent_education_scores_melted =
parent_education_scores.melt(id_vars='parent_education',
var_name='Subject', value_name='Average Score')
sns.barplot(x='parent_education', y='Average Score', hue='Subject',
data=parent_education_scores_melted, palette='magma')
plt.title('Average Scores by Parental Education Level')
plt.xlabel('Parental Education')
plt.ylabel('Average Score')
plt.ylim(0, 100)
plt.xticks(rotation=45, ha='right')
plt.legend(title='Subject')
plt.tight_layout()
plt.show()
```

