

KSR College Of Engineering , Tiruchengode

Department of Information Technology

NALAIYA THIRAN

AI ASSESMENT- 1

```
{
  "nbformat": 4,
  "nbformat_minor": 0,
  "metadata": {
    "colab": {
      "provenance": [],
      "collapsed_sections": []
    },
    "kernelspec": {
      "name": "python3",
      "display_name": "Python 3"
    },
    "language_info": {
      "name": "python"
    }
  },
  "cells": [
    {
      "cell_type": "markdown",
```

```
"source": [  
  "# Basic Python"  
],  
"metadata": {  
  "id": "McSxJAwcOdZ1"  
}  
},  
{  
  "cell_type": "markdown",  
  "source": [  
    "## 1. Split this string"  
  ],  
  "metadata": {  
    "id": "CU48hgo4Owz5"  
  }  
},  
{  
  "cell_type": "code",  
  "source": [  
    "s = \"Hi there Sam!\\n\",  
    "\\n",  
    "\\n",  
    "n=s.split()\\n",  
    "print(n)"  
  ],
```

```
"metadata": {
  "id": "6mGVa3SQYLkb",
  "outputId": "3dd622bb-5067-442f-976d-877f2a16490b",
  "colab": {
    "base_uri": "https://localhost:8080/"
  }
},
"execution_count": null,
"outputs": [
  {
    "output_type": "stream",
    "name": "stdout",
    "text": [
      "['Hi', 'there', 'Sam!']\n"
    ]
  }
],
{
  "cell_type": "markdown",
  "source": [
    "## 2. Use .format() to print the following string. \n",
    "\n",
    "### Output should be: The diameter of Earth is 12742 kilometers."
  ],
}
```

```
"metadata": {
  "id": "GH1QBn8HP375"
}
},
{
  "cell_type": "code",
  "source": [
    "planet = \"Earth\\n\",
    "diameter = 12742\\n",
    "\\n",
    "\\n",
    "\\n",
    "star = \"The diameter of {p} is {d} kilometers \\n\",
    "print(star.format(p=planet,d=diameter))"
  ],
  "metadata": {
    "id": "_ZHoml3kPqic",
    "outputId": "69299b93-c410-425c-b48a-5b17fbea4a14",
    "colab": {
      "base_uri": "https://localhost:8080/"
    }
  },
  "execution_count": null,
  "outputs": [
    {
```

```

    "output_type": "stream",

    "name": "stdout",

    "text": [

        "The diameter of Earth is 12742 kilometers \n"

    ]

}

],

{

    "cell_type": "markdown",

    "source": [

        "## 3. In this nest dictionary grab the word \"hello\""

    ],

    "metadata": {

        "id": "KE74ZEwkRExZ"

    }

},

{

    "cell_type": "code",

    "source": [

        "d = {'k1':[1,2,3,{'tricky':['oh','man','inception',{'target':[1,2,3,'hello']}]}]}\n",

        "\n",

        "\n",

        "\n",

        "d['k1'][3]['tricky'][3]['target'][3]"
    ]

```

```
],
"metadata": {
  "id": "fcVwbCc1QrQI",
  "outputId": "a76f16e9-1e1d-4410-d528-a24a27a34cc1",
  "colab": {
    "base_uri": "https://localhost:8080/",
    "height": 36
  }
},
"execution_count": null,
"outputs": [
  {
    "output_type": "execute_result",
    "data": {
      "text/plain": [
        "'hello'"
      ],
      "application/vnd.google.colaboratory.intrinsic+json": {
        "type": "string"
      }
    },
    "metadata": {},
    "execution_count": 4
  }
]
```

```
},  
  
{  
  "cell_type": "markdown",  
  "source": [  
    "# Numpy"  
  ],  
  "metadata": {  
    "id": "bw0vVp-9ddjv"  
  }  
},  
  
{  
  "cell_type": "code",  
  "source": [  
    "import numpy as np"  
  ],  
  "metadata": {  
    "id": "LLiE_TYrhA1O"  
  },  
  "execution_count": null,  
  "outputs": []  
},  
  
{  
  "cell_type": "markdown",  
  "source": [  
    "## 4.1 Create an array of 10 zeros? \n",
```

```
### 4.2 Create an array of 10 fives?"
```

```
],
```

```
"metadata": {
```

```
  "id": "wOg8hinbgx30"
```

```
}
```

```
},
```

```
{
```

```
  "cell_type": "code",
```

```
  "source": [
```

```
    "a=np.zeros(10)\n",
```

```
    "print(a)"
```

```
],
```

```
"metadata": {
```

```
  "id": "NHrirmgCYXvU",
```

```
  "outputId": "fe1809ab-9daa-4453-f989-2e27e301d27a",
```

```
  "colab": {
```

```
    "base_uri": "https://localhost:8080/"
```

```
}
```

```
},
```

```
"execution_count": null,
```

```
"outputs": [
```

```
{
```

```
  "output_type": "stream",
```

```
  "name": "stdout",
```

```
  "text": [
```



```
    "[0. 0. 0. 0. 0. 0. 0. 0. 0.]\n"
  ]
}
]
},
{
  "cell_type": "code",
  "source": [
    "b=np.ones(10)*5\n",
    "print(b)"
  ],
  "metadata": {
    "id": "e4005lsTYXxx",
    "outputId": "b74212f9-7a00-4ee8-df92-6b7bfd15eb34",
    "colab": {
      "base_uri": "https://localhost:8080/"
    }
  },
  "execution_count": null,
  "outputs": [
    {
      "output_type": "stream",
      "name": "stdout",
      "text": [
        "[5. 5. 5. 5. 5. 5. 5. 5. 5.]\n"
      ]
    }
  ]
}
```

```
]
}
]
},
{
  "cell_type": "markdown",
  "source": [
    "## 5. Create an array of all the even integers from 20 to 35"
  ],
  "metadata": {
    "id": "gZHHdUBvrMX4"
  }
},
{
  "cell_type": "code",
  "source": [
    "a=np.arange(20,35,2)\n",
    "print(a)"
  ],
  "metadata": {
    "id": "oAl2tbU2Yag-",
    "outputId": "138baaf0-d91d-4009-e79e-f992861a4b7c",
    "colab": {
      "base_uri": "https://localhost:8080/"
    }
  }
}
```

```
},
"execution_count": null,
"outputs": [
  {
    "output_type": "stream",
    "name": "stdout",
    "text": [
      "[20 22 24 26 28 30 32 34]\n"
    ]
  }
],
},
{
  "cell_type": "markdown",
  "source": [
    "## 6. Create a 3x3 matrix with values ranging from 0 to 8"
  ],
},
{
  "cell_type": "code",
  "source": [
    "a=np.arange(0,9).reshape(3,3)\n",
```

```
"print(a)"
],
"metadata": {
  "id": "tOIEVH7BYceE",
  "outputId": "598583a2-b037-408e-da3f-8491eec773fe",
  "colab": {
    "base_uri": "https://localhost:8080/"
  }
},
"execution_count": null,
"outputs": [
  {
    "output_type": "stream",
    "name": "stdout",
    "text": [
      "[[0 1 2]\n",
      "[3 4 5]\n",
      "[6 7 8]]\n"
    ]
  }
],
},
{
  "cell_type": "markdown",
  "source": [
```

```

    "## 7. Concatenate a and b \n",

    "## a = np.array([1, 2, 3]), b = np.array([4, 5, 6])"

],

"metadata": {

    "id": "hQ0dnhAQuU_p"

}

},

{

    "cell_type": "code",

    "source": [

        "a=np.array ([1,2,3])\n",

        "b=np.array([4,5,6])\n",

        "c=np.concatenate((a,b),axis=0)\n",

        "print(c)\n",

        "\n"

    ],

    "metadata": {

        "id": "rAPSw97aYfEO",

        "outputId": "900956cb-76f9-4857-8584-4baee03c809f",

        "colab": {

            "base_uri": "https://localhost:8080/"

        }

    },

    "execution_count": null,

    "outputs": [

```

```
{
  "output_type": "stream",
  "name": "stdout",
  "text": [
    "[1 2 3 4 5 6]\n"
  ]
}

],

{
  "cell_type": "markdown",
  "source": [
    "# Pandas"
  ],
  "metadata": {
    "id": "dIPEY9DRwZga"
  }
},

{
  "cell_type": "markdown",
  "source": [
    "## 8. Create a dataframe with 3 rows and 2 columns"
  ],
  "metadata": {
    "id": "ijoYW51zwr87"
```

```
}  
  
,  
  
{  
  "cell_type": "code",  
  "source": [  
    "import pandas as pd\n",  
  ],  
  "metadata": {  
    "id": "T5OxJRZ8uvR7"  
  },  
  "execution_count": null,  
  "outputs": []  
},  
  
{  
  "cell_type": "code",  
  "source": [  
    "data=[['muni',10],['sai',20],['nathan',30]]\n",  
    "a=pd.DataFrame(data,columns=['Name','Age'])\n",  
    "print(a)"  
  ],  
  "metadata": {  
    "id": "xNpl_XXoYhs0",  
    "outputId": "2f2825df-6eb4-4fc9-9ed2-3e20dc2e49f8",  
    "colab": {  
      "base_uri": "https://localhost:8080/"
```

```
}  
  
,  
  
"execution_count": null,  
  
"outputs": [  
  
  {  
  
    "output_type": "stream",  
  
    "name": "stdout",  
  
    "text": [  
  
      "  Name Age\n",  
  
      "0  muni 10\n",  
  
      "1  sai 20\n",  
  
      "2  nathan 30\n"  
  
    ]  
  
  }  
  
]  
  
,  
  
{  
  
  "cell_type": "markdown",  
  
  "source": [  
  
    "## 9. Generate the series of dates from 1st Jan, 2023 to 10th Feb, 2023"  
  
  ],  
  
  "metadata": {  
  
    "id": "UXSmdNclyJQD"  
  
  }  
  
},
```



```
{
  "cell_type": "code",
  "source": [
    "from datetime import date,timedelta\n",
    "sdate =date(2023,1,1)\n",
    "edate=date(2023,2,11)\n",
    "[sdate+timedelta(days=x)for x in range((edate-sdate).days)]"
  ],
  "metadata": {
    "id": "dgyC0JhVYl4F",
    "outputId": "2552a2b5-8926-4bad-e3a7-932ed999ac39",
    "colab": {
      "base_uri": "https://localhost:8080/"
    }
  },
  "execution_count": 3,
  "outputs": [
    {
      "output_type": "execute_result",
      "data": {
        "text/plain": [
          "[datetime.date(2023, 1, 1),\n",
          " datetime.date(2023, 1, 2),\n",
          " datetime.date(2023, 1, 3),\n",
          " datetime.date(2023, 1, 4),\n"
```

" datetime.date(2023, 1, 5),\n",
" datetime.date(2023, 1, 6),\n",
" datetime.date(2023, 1, 7),\n",
" datetime.date(2023, 1, 8),\n",
" datetime.date(2023, 1, 9),\n",
" datetime.date(2023, 1, 10),\n",
" datetime.date(2023, 1, 11),\n",
" datetime.date(2023, 1, 12),\n",
" datetime.date(2023, 1, 13),\n",
" datetime.date(2023, 1, 14),\n",
" datetime.date(2023, 1, 15),\n",
" datetime.date(2023, 1, 16),\n",
" datetime.date(2023, 1, 17),\n",
" datetime.date(2023, 1, 18),\n",
" datetime.date(2023, 1, 19),\n",
" datetime.date(2023, 1, 20),\n",
" datetime.date(2023, 1, 21),\n",
" datetime.date(2023, 1, 22),\n",
" datetime.date(2023, 1, 23),\n",
" datetime.date(2023, 1, 24),\n",
" datetime.date(2023, 1, 25),\n",
" datetime.date(2023, 1, 26),\n",
" datetime.date(2023, 1, 27),\n",
" datetime.date(2023, 1, 28),\n",
" datetime.date(2023, 1, 29),\n",

```

" datetime.date(2023, 1, 30),\n",
" datetime.date(2023, 1, 31),\n",
" datetime.date(2023, 2, 1),\n",
" datetime.date(2023, 2, 2),\n",
" datetime.date(2023, 2, 3),\n",
" datetime.date(2023, 2, 4),\n",
" datetime.date(2023, 2, 5),\n",
" datetime.date(2023, 2, 6),\n",
" datetime.date(2023, 2, 7),\n",
" datetime.date(2023, 2, 8),\n",
" datetime.date(2023, 2, 9),\n",
" datetime.date(2023, 2, 10)]"
]
},
"metadata": {},
"execution_count": 3
}
]
},
{
"cell_type": "markdown",
"source": [
"## 10. Create 2D list to DataFrame\n",
"\n",
"lists = [[1, 'aaa', 22],\n",

```

```

        "    [2, 'bbb', 25],\n",
        "    [3, 'ccc', 24]]"
    ],
    "metadata": {
        "id": "ZizSetD-y5az"
    }
},
{
    "cell_type": "code",
    "source": [
        "lists = [[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]\n",
        "\n",
        "\n",
        "a=pd.DataFrame(lists,columns=['Number','FName','Age'])\n",
        "print(a)"
    ],
    "metadata": {
        "id": "_XMC8aEt0IIB",
        "outputId": "2413aaed-1f46-4e42-abc6-d4b05a1eee4f",
        "colab": {
            "base_uri": "https://localhost:8080/"
        }
    },
    "execution_count": null,
    "outputs": [

```

```
{  
  "output_type": "stream",  
  "name": "stdout",  
  "text": [  
    "  Number FName Age\n",  
    "0    1  aaa  22\n",  
    "1    2  bbb  25\n",  
    "2    3  ccc  24\n"  
  ]  
}  
]  
}  
]
```