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!\""
      ],
      "metadata": {
        "id": "s07c7JK7Oqt-"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "n=s.split()\n",
        "print(n)"
      ],
      "metadata": {
```

```
    "id": "6mGVa3SQYLkb",
    "colab": {
      "base_uri": "https://localhost:8080/"
    },
    "outputId": "4aac7685-8efb-482d-b47b-9cb29a6dc618"
  },
  "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"
    ],
    "metadata": {
      "id": "_ZHoml3kPqic"
    },
    "execution_count": null,
    "outputs": []
  },
  {
    "cell_type": "code",
    "source": [
      "print('The diameter of {} is {} kilometers.'.format(planet, diameter))"
    ],
    "metadata": {
      "id": "HyRyJv6CYPb4",
      "colab": {
        "base_uri": "https://localhost:8080/"
      },
      "outputId": "5bd7def2-98e7-4a5e-8b06-3c802ad5bed3"
    },
    "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']}]}]}"
      ],
      "metadata": {
        "id": "fcVwbCc1QrQI"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "d['k1'][3]['tricky'][3]['target'][3]"
      ],
      "metadata": {
        "id": "MvbkMZpXYRaw",
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 35

```
      },
      "outputId": "99eab825-c5e4-4e10-9ac1-e0fc7699b1bc"
    },
    "execution_count": null,
    "outputs": [
      {
        "output_type": "execute_result",
        "data": {
          "text/plain": [
            "'hello'"
          ],
          "application/vnd.google.colaboratory.intrinsic+json": {
            "type": "string"
          }
        },
        "metadata": {},
        "execution_count": 6
      }
    ]
  },
  {
    "cell_type": "markdown",
    "source": [
      "# Numpy"
    ],
    "metadata": {
      "id": "bw0vVp-9ddjv"
    }
  },
  {
    "cell_type": "code",
```

```json
    "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": [
     "array=np.zeros(10)\n",
     "print(\"An array of 10 zeros\")\n",
     "print(array)"
    ],
    "metadata": {
     "id": "NHrirmgCYXvU",
     "colab": {
      "base_uri": "https://localhost:8080/"
     },
```

```
      "outputId": "f77de545-d11b-4145-fc88-0e2f395979cf"
    },
    "execution_count": null,
    "outputs": [
      {
        "output_type": "stream",
        "name": "stdout",
        "text": [
          "An array of 10 zeros\n",
          "[0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]\n"
        ]
      }
    ]
  },
  {
    "cell_type": "code",
    "source": [
      "array=np.ones(10)*5\n",
      "print(\"An array of 10 fives\")\n",
      "print(array)"
    ],
    "metadata": {
      "id": "e4005lsTYXxx",
      "colab": {
        "base_uri": "https://localhost:8080/"
      },
      "outputId": "fd31a750-af9f-4164-9217-e29a9af2e2c9"
    },
    "execution_count": null,
    "outputs": [
      {
```

```json
      "output_type": "stream",
      "name": "stdout",
      "text": [
        "An array of 10 fives\n",
        "[5. 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": "oAI2tbU2Yag-",
    "colab": {
      "base_uri": "https://localhost:8080/"
    },
    "outputId": "36166281-fd5e-4b70-83ee-0a92db38f47f"
  },
```

```json
    "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"
    ],
    "metadata": {
     "id": "NaOM308NsRpZ"
    }
   },
   {
    "cell_type": "code",
    "source": [
     "x=np.arange(0,9).reshape(3,3)\n",
     "print(x)"
    ],
    "metadata": {
     "id": "tOlEVH7BYceE",
     "colab": {
      "base_uri": "https://localhost:8080/"
     },
```

```
      "outputId": "f61cf420-5f06-490a-adc2-9b614b912b17"
    },
    "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",
     "np.concatenate((a, b,), axis=0, out=None)"
```

```
    ],
    "metadata": {
      "id": "rAPSw97aYfE0",
      "colab": {
        "base_uri": "https://localhost:8080/"
      },
      "outputId": "9d787b9f-1f8b-4a5f-d427-1a85081f72dd"
    },
    "execution_count": null,
    "outputs": [
      {
        "output_type": "execute_result",
        "data": {
          "text/plain": [
            "array([1, 2, 3, 4, 5, 6])"
          ]
        },
        "metadata": {},
        "execution_count": 18
      }
    ]
  },
  {
    "cell_type": "markdown",
    "source": [
      "# Pandas"
    ],
    "metadata": {
      "id": "dlPEY9DRwZga"
    }
  },
```

```
{
  "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=[ [ 'Hari' , 55], ['vamsi' , 20], [ 'sai' , 30] ]\n",
    "a=pd . DataFrame (data, columns=[ 'Name' , 'Age' ])\n",
    "\n",
    "print(a)\n"
  ],
  "metadata": {
    "id": "xNpI_XXoYhs0",
    "colab": {
```

```
      "base_uri": "https://localhost:8080/"
     },
     "outputId": "6604a2bb-51d5-4711-cda3-cd422626c127"
    },
    "execution_count": null,
    "outputs": [
     {
      "output_type": "stream",
      "name": "stdout",
      "text": [
       "    Name  Age\n",
       "0   Hari   55\n",
       "1  vamsi   20\n",
       "2    sai   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",
```

    "\n",

  "sdate = date(2023,1,1)   \n",

  "edate = date(2023,2,11)   \n",

  "\n",

  "[sdate+timedelta(days=x) for x in range((edate-sdate).days)]"

 ],

 "metadata": {

  "id": "dgyC0JhVYl4F",

  "colab": {

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

  },

  "outputId": "517f0574-4b15-40ec-ccdb-4858e7fbdbe0"

 },

 "execution_count": null,

 "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)]"
   ]
  },
```

```json
      "metadata": {},
      "execution_count": 22
    }
   ]
 },
 {
  "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]]"
  ],
  "metadata": {
   "id": "_XMC8aEt0llB"
  },
  "execution_count": null,
  "outputs": []
 },
 {
  "cell_type": "code",
```

```json
    "source": [
     "lists=[[1, 'aaa', 22], [2, 'bbb', 25], [3, 'ccc', 24]]\n",
     "df=pd.DataFrame(lists, columns=[ 'Number' , 'FName' , 'Age' ])\n",
     "print (df)"
    ],
    "metadata": {
     "id": "knH76sDKYsVX",
     "colab": {
      "base_uri": "https://localhost:8080/"
     },
     "outputId": "59a459c2-7e88-4809-e8c7-b81e73f47649"
    },
    "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"
      ]
     }
    ]
   }
  ]
}
```