

```
{
  "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": "CU48hgo40wz5"
      },
      "cell_type": "code",
      "source": [
        "s = \"Hi there Sam!\""
      ],
      "metadata": {
        "id": "s07c7JK70qt-"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        ""
      ],
      "metadata": {
        "id": "6mGVa3SQYLkb"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "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": [
        ""
      ],
      "metadata": {
        "id": "HyRyJv6CYPb4"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "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": [
        ""
      ],
      "metadata": {
        "id": "MvbkMZpXYRaw"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "source": [
        "# Numpy"
      ],
      "metadata": {
        "id": "bw0vVp-9ddjv"
      },
      "cell_type": "code",
      "source": [
        "import numpy as np"
      ],
      "metadata": {
        "id": "LLiE_TYrhA10"
      },
      "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": [
        ""
      ],
      "metadata": {
        "id": "NHhirmgCYXvU"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        ""
      ],
      "metadata": {
        "id": "e4005lsTYXxx"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "source": [
        "## 5. Create an array of all the even integers from 20 to 35"
      ],
      "metadata": {
        "id": "gZHHdUBvrMX4"
      },
      "cell_type": "code",
      "source": [
        ""
      ],
      "metadata": {
        "id": "oAl2tbU2Yag-"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "source": [
        "## 6. Create a 3x3 matrix with values ranging from 0 to 8"
      ],
      "metadata": {
        "id": "NaOM308NsRpZ"
      },
      "cell_type": "code",
      "source": [
        ""
      ],
      "metadata": {
        "id": "tOIEVH7BYceE"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "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": [
        ""
      ],
      "metadata": {
        "id": "rAPSw97aYfE0"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "source": [
        "# Pandas"
      ],
      "metadata": {
        "id": "dLPEY9DRwZga"
      },
      "cell_type": "code",
      "source": [
        ""
      ],
      "metadata": {
        "id": "ijoYW51zwr87"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        ""
      ],
      "metadata": {
        "id": "T50xJRZ8uvR7"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        ""
      ],
      "metadata": {
        "id": "xNpl_XXoYhs0"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "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": [
        ""
      ],
      "metadata": {
        "id": "dgyC0JhVYl4F"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        ""
      ],
      "metadata": {
        "id": "ZizSetD-y5az"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        ""
      ],
      "metadata": {
        "id": "_XMC8aEt0lIB"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        ""
      ],
      "metadata": {
        "id": "knH76sDKYsVX"
      },
      "execution_count": null,
      "outputs": []
    }
  ]
}
```