

\rtf1\ansi\ansicpg1252\cocoartf2818
\cocoatextscaling0\cocoaplatform0{\fonttbl{\f0\froman\fcharset0 Times-Bold;\f1\froman\fcharset0 Times-Roman;\f2\modern\fcharset0 Courier;
}
{\colortbl;\red255\green255\blue255;\red0\green0\blue0;\red0\green0\blue233;\red218\green76\blue12;
\red87\green96\blue106;\red6\green33\blue79;\red7\green68\blue184;}
{*\expandedcolortbl;;\cssrgb\c0\c0\c0;\cssrgb\c0\c0\c93333;\cssrgb\c89020\c38431\c3529;
\cssrgb\c41569\c45098\c49020;\cssrgb\c1176\c18431\c38431;\cssrgb\c0\c36078\c77255;}
{*\listtable{\list\listtemplateid1\listhybrid{\listlevel\levelnfc23\levelnfcn23\leveljc0\leveljcn0\levelfollow0\levelstartat1\levelspace360\levelindent0{*\levelmarker \{disc\}}
{\leveltext\leveltemplateid1\'01\uc0\u8226 ;}{\levelnumbers;}\fi-360\li720\lin720 }{\listname ;}\listid1}
{\list\listtemplateid2\listhybrid{\listlevel\levelnfc23\levelnfcn23\leveljc0\leveljcn0\levelfollow0\levelstartat1\levelspace360\levelindent0{*\levelmarker \{disc\}}
{\leveltext\leveltemplateid101\'01\uc0\u8226 ;}{\levelnumbers;}\fi-360\li720\lin720 }{\listname ;}\listid2}
{\list\listtemplateid3\listhybrid{\listlevel\levelnfc0\levelnfcn0\leveljc0\leveljcn0\levelfollow0\levelstartat1\levelspace360\levelindent0{*\levelmarker \{decimal\}}
{\leveltext\leveltemplateid201\'01\'00;}{\levelnumbers\'01;}\fi-360\li720\lin720 }{\listname ;}\listid3}}
{*\listoverridetable{\listoverride\listid1\listoverridecount0\ls1}
{\listoverride\listid2\listoverridecount0\ls2}{\listoverride\listid3\listoverridecount0\ls3}}
\defstab720
\pard\pardefstab720\sa428\partightenfactor0

\fb\fs64 \cf0 \expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Chapter 1: Introduction to Python\
\pard\pardefstab720\sa398\partightenfactor0

\fs48 \cf0 1.1 - Python Basics\
\pard\pardefstab720\sa320\partightenfactor0

\fb\fb\fs32 \cf0 Python is a high-level programming language that is known for its simplicity and readability. It is a great language for beginners, but also has the depth to be used for advanced scientific and data analysis.\br/>
The first thing you need to do before you can start coding is to install Python. You can download the latest version of Python from the official website ({\field{*\fldinst{HYPERLINK "https://www.python.org/"}{\fldrslt \cf3 \expnd0\expndtw0\kerning0
\ul \ulc3 \outl0\strokewidth0 \strokec3 https://www.python.org/}}).\br/>
Once you have Python installed, you can write Python scripts using any text editor, but it\'92s often easier to use a dedicated Python Integrated Development Environment (IDE) like PyCharm or Jupyter Notebook.\br/>
To run a Python script, save your code with a .py extension and run it through the Python interpreter in your command line interface.\br/>
\pard\pardefstab720\sa374\partightenfactor0

\fb\fb\fs37\fsmilli18720 \cf0 Hello, World!\br/>
\pard\pardefstab720\sa320\partightenfactor0

\fb\fb\fs32 \cf0 The traditional first program to write in any language is a \'93Hello, World!\'94 program. This is a simple program that prints the phrase \'93Hello, World!\'94 to the console.\br/>
In Python, this is as easy as:\br/>
\pard\pardefstab720\partightenfactor0

\fb\fs26 \cf4 \strokec4 print\cf5 \strokec5 (\cf6 \strokec6 "Hello, World!"\cf5 \strokec5)\
\pard\pardefstab720\sa398\partightenfactor0

\fb\fb\fs48 \cf0 \strokec2 1.2 - Variables and Types\
\pard\pardefstab720\sa320\partightenfactor0

\fb\fb\fs32 \cf0 In Python, you don\'92t have to declare the type of a variable when you create it; Python automatically determines the type. Here are a few examples:\br/>
\pard\pardefstab720\partightenfactor0

```
\f2\fs26 \cf5 \strokec5 name = \cf6 \strokec6 "Alice"\cf5 \strokec5 # a string\
age = \cf7 \strokec7 30\cf5 \strokec5 # an integer\
height = \cf7 \strokec7 5.5\cf5 \strokec5 # a floating-point number\
\pard\pardeftab720\sa320\partightenfactor0

\f1\fs32 \cf0 \strokec2 Python supports several types of data, but the most common are:\
\pard\tx220\tx720\pardeftab720\li720\fi-720\partightenfactor0
\ls1\ilvl0
\f2\fs26 \cf0 \kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 int
\f1\fs32 : an integer, e.g., 1, 2, 3\
\ls1\ilvl0
\f2\fs26 \kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 float
\f1\fs32 : a floating-point (decimal) number, e.g., 1.0, 1.5, 3.14\
\ls1\ilvl0
\f2\fs26 \kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 str
\f1\fs32 : a string (text), e.g., '93hello'94, '93world'94\
\ls1\ilvl0
\f2\fs26 \kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 bool
\f1\fs32 : a boolean, which can be True or False\
\pard\pardeftab720\sa320\partightenfactor0
\cf0 You can use the
\f2\fs26 type()
\f1\fs32 function to check the type of a variable:\
\pard\pardeftab720\partightenfactor0

\f2\fs26 \cf4 \strokec4 print\cf5 \strokec5 (\cf4 \strokec4 type\cf5 \strokec5 (name)) # <class
'str'>\
\pard\pardeftab720\sa398\partightenfactor0

\f0\b\fs48 \cf0 \strokec2 1.3 - Basic Arithmetic\
\pard\pardeftab720\sa320\partightenfactor0

\f1\b0\fs32 \cf0 Python supports the basic arithmetic operations you would expect:\
\pard\tx220\tx720\pardeftab720\li720\fi-720\partightenfactor0
\ls2\ilvl0\cf0 \kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Addition (+)\
\ls2\ilvl0\kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Subtraction (-)\
\ls2\ilvl0\kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Multiplication (*)\
\ls2\ilvl0\kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Division (/)\
\ls2\ilvl0\kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Exponentiation (**)\
\ls2\ilvl0\kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Modulus (%)\
\ls2\ilvl0\kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext \uc0\u8226
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Floor division (//)\
\pard\pardeftab720\sa320\partightenfactor0
\cf0 Here'92s how you might use them:\
```

```

\pard\pardefstab720\partightenfactor0

\f2\fs26 \cf5 \strokec5 x = \cf7 \strokec7 10\cf5 \strokec5 \
y = \cf7 \strokec7 2\cf5 \strokec5 \
\
\pard\pardefstab720\partightenfactor0
\cf4 \strokec4 print\cf5 \strokec5 (x + y) # 12\
\cf4 \strokec4 print\cf5 \strokec5 (x - y) # 8\
\cf4 \strokec4 print\cf5 \strokec5 (x * y) # 20\
\cf4 \strokec4 print\cf5 \strokec5 (x / y) # 5.0\
\cf4 \strokec4 print\cf5 \strokec5 (x ** y) # 100\
\cf4 \strokec4 print\cf5 \strokec5 (x % y) # 0\
\cf4 \strokec4 print\cf5 \strokec5 (x // y) # 5\
\pard\pardefstab720\sa398\partightenfactor0

\f0\b\fs48 \cf0 \strokec2 Exercises\
\pard\tx220\tx720\pardefstab720\li720\fi-720\partightenfactor0
\ls3\ilvl0
\f1\b0\fs32 \cf0 \kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext 1
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Print your name using the
\f2\fs26 print()
\f1\fs32 function.\
\ls3\ilvl0\kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext 2
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Create variables representing your age and height, and print
them.\
\ls3\ilvl0\kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext 3
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Do some basic arithmetic: add, subtract, multiply, and divide some
numbers.\
\ls3\ilvl0\kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext 4
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Try using the modulus and exponentiation operators.\
\ls3\ilvl0\kerning1\expnd0\expndtw0 \outl0\strokewidth0 {\listtext 5
}\expnd0\expndtw0\kerning0
\outl0\strokewidth0 \strokec2 Find out the types of all the variables you've created using
the
\f2\fs26 type()
\f1\fs32 function.\
\pard\pardefstab720\sa320\partightenfactor0
\cf0 In the next chapter, we'll delve deeper into data types and start looking at Python's
control flow mechanisms, like
\f2\fs26 if
\f1\fs32 statements and loops.}

```