**Tipos de dados (padrão da linguagem):**

Texto – str;

Numéricos - int, float, complex;

Sequência – list, tuple, range;

Mapa – dict (chave: valor);

Coleção – set, fronzenset;

Booleano – bool;

Binário – bytes, bytearray, memoryview;

**Interpretador Python:** programar e ir vendo o resultado no terminal

Digitar “python” ou “python -i nome\_do\_programa.py” no terminal. Para sair digitar “exit()”

**Dir:** Sem argumento, retorna a lista de nomes no escopo local atual. Com um argumento, retorna uma lista de atributos válidos para o objeto. Exemplo

- dir()

- dir(100): lista de métodos dentro da classe int

**Help:** Invoca o sistema de ajuda integrado. É possível fazer buscas em modo interativo ou informar por parâmetro qual o nome do módulo, função, classe, método ou variável. Exemplo:

- help(): você consegue buscar o método e ele retorna todos os métodos e utilidades daquilo que foi buscado;

-help(100): aqui vai trazer as informações sobre os métodos da classe int

**VARIÁVEIS E CONSTANTES**

**Variáveis:** seu valor muda ao longo do programa. Definição: variável = valor

**Constantes:** seu valor não muda, uma vez atribuído um valor, ele sempre será. Exemplo: um programa sobre dados pessoais, o tipo sanguíneo será um valor constante. Definição: utiliza-se o nome da variável, por convenção, deve ser escrito todo em maiúsculo, VARIAVEL = valor.

**Boas práticas:**

- Padrão de nomes deve ser o snake case: substituir espaços pelo *underline*;

- Escolher nomes sugestivos;

- Nome de constantes todo em maiúsculo.

**CONVERSÃO DE TIPOS**

Guido Van Rossum – 1989: Linguagem intuitiva, código aberta

-Tipagem dinâmica e forte;

- Multiplataforma e multiparadigma(paradigma procedural, funcional, orientado a objetos...);

- Comunidade gigante e ativa;

- Curva de aprendizagem baixa