Python

* Interpretersprache (Zwischendateien (mpc), Zwischencode)
* Plattformunabhängigkeit
* umfangreiche Standardbibliotheken
* PSF-Lizenz (open source, Python Code überall einbauen, frei zum Weiterbearbeiten), offener als GNU
* Rapid Prototyping: schnell und sofort Prototypen bauen, weil man kann auch funktional und iterativ programmieren (man muss kein Klassen schreiben, sondern man kann alles auf eine Seite schreiben, …)
* Garbage Collection: Daten, die für eine längere Zeit nicht verwendet werden, werden automatisch gelöscht; Verfahren zur Speicherbereinigung und Wiederfreigabe von belegtem Speicherplatz, der aber vom jeweiligen Computerprogramm nicht mehr benötigt wird
* Instanz/Referenz/GarbageCollection/del/Referenzzähler
* keine Datentypen (type): während der Laufzeit (dynamisch) werden Datentypen klar
* funktionale und objektorientierte Programmierung möglich
* Python2/Python3 (in P3 mit Abwärtskompabilität gebrochen) 🡪 immer beachten ob Python 2 oder 3
* <https://www.learnpython.org/>
* <https://www.programiz.com/python-programming>

**Unterschiede zu Java:**

* Datentypen dynamisch vergeben
* Einrückungen und ‘:‘ statt geschweifte, … Klammern
* keine main-Methode 🡪 man kann alles in einer Wurst schreiben (ob es gut istm ist eine andere Frage)

**Binary Search:**

* <https://www.programiz.com/dsa/binary-search>

**Allgemein:**

* Referenz = Addresse die auf den Value zeigt
* Call by Reference: wir übergeben nur die Referenz zum Value
* Call by Value: wir übergeben wirklich den Wert bzw. Inhalt