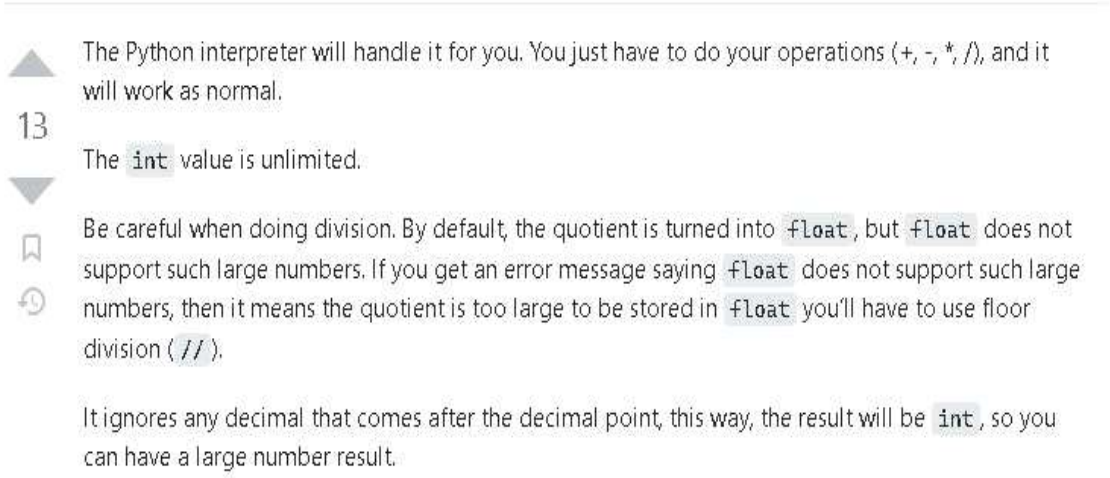
DIFF BetDeclarativ & Imparaive & Procedtural Writing Code ??ween

In coding, declarative programming refers to code that is concerned with higher levels of abstraction. It describes what you want the program to achieve without describing its control flow. Imperative programming, on the other hand, refers to code that is concerned with lower levels of abstraction. It describes how you want the program to achieve the desired result by specifying how it is to be done. Procedural programming is a subset of imperative programming which utilizes subroutines[158].

Declarative programming is about what a program does, while imperative programming is about how a program works. Imperative programming is easier to reason about for beginners, while declarative programming allows us to write more readable code that reflects what exactly we want to see[27].

How Can Python Deal with Big Nums ??



13

The Python interpreter will handle it for you. You just have to do your operations (+, -, *, /), and it will work as normal.

The `int` value is unlimited.

Be careful when doing division. By default, the quotient is turned into `float`, but `float` does not support such large numbers. If you get an error message saying `float` does not support such large numbers, then it means the quotient is too large to be stored in `float` you'll have to use floor division (`//`).

It ignores any decimal that comes after the decimal point, this way, the result will be `int`, so you can have a large number result.

Diff Between HEAP & STACK ??

In computer programming, *stack* and *heap* are two important concepts. Stack is a linear data structure that stores values temporarily. It can store only local or temporary variables. On the other hand, Heap is a hierarchical data structure that is typically used for dynamic memory allocation. It acts as a pool of storage area that allocates memory randomly (Dynamic memory allocation)[36].

Stack memory will never become fragmented whereas Heap memory can become fragmented as blocks of memory are first allocated and then freed[1]. Stack accesses local variables only while Heap allows you to access variables globally[1].

NOT Case Sinsetive PL examples ??

- **FOrtran**
- **BASIC**
- **Pascal**
- **Ada**
- **SQL**

Garpage Collection EXs (Autu & Not-Autu) ??

Programming languages that have automatic garbage collection include

- **C#**
- **Java**
- **Python**
- **Ruby**

and many others. Programming languages that do not have automatic garbage collection include

- **C**
- **C++**
- **Pascal**