

# Function calls and stack frames

- each function call in Python creates a stack frame which stores the local variables and parameters of the function
- when the function returns, the return value is left on top of the stack for the calling function to access
- example call is `toStr(5,2)`
- top frame will be leave '1' on stack after call
- middle frame will leave '10' on stack after call
- bottom frame will be leave '101' on stack after call

```
def toStr(n,base):  
    characters = '0123456789ABCDEF'  
    if n < base:  
        return characters[n]  
    return toStr(n//base,base) + \  
           characters[n%base]
```

