computer_builds.models.resources

▸ View Source

class Resources: ▸ View Source

Resources base class

Resources(name, manufacturer, total, allocated) ▸ View Source

Args: name (str): user-friendly name of resource instance (e.g. `Intel Core i9-9900K`) manufacturer (str): resource instance manufacturer (e.g. `Nvidia`) total (int): inventory total (how many are in the inventory pool) allocated (int): number allocated (how many are already in use), must be smaller than total

total

Returns: total (int): total amount of resource

allocated

Returns: allocated (int): amount of allocated resource

name

Returns: name(str):resource name

manufacturer

Returns: manufacturer(str):resource manufacturer

def claim(self, n): ▸ View Source

Claim n inventory items if available Args: n(int): number of items to claim Returns: None

def freeup(self, n): ▸ View Source

Free up n items if possible Args: n(int): number of items to free up Returns: None

def died(self, n): ▸ View Source

Subtract n items from total and allocated (if possible) Args: n(int): number of items to subtract Returns: None

def purchased(self, n): ▸ View Source

Add n items to the total Args: n(int): number of items to add Returns: None

category

Returns: category(str):resource category

class CPU(Resources): ▸ View Source

Resources subclass class for tracking CPU inventory

CPU(name, manufacturer, total, allocated, cores, socket, power_watts) ▸ View Source

Args: name (str): user-friendly name of resource instance (e.g. `Intel Core i9-9900K`) manufacturer (str): resource instance manufacturer (e.g. `Nvidia`) total (int): inventory total (how many are in the inventory pool) allocated (int): number allocated (how many are already in use), must be smaller than total cores (int): number of cores is even and ranges from 2 to 100 (e.g. `8`) socket (str): socket (e.g. `AM4`) power_watts (int): number of watts (e.g. `94`)

cores

Returns: cores (int): number of cores is even and ranges from 2 to 100 (e.g. `8`)

socket

Returns: socket (str): socket (e.g. `AM4`)

power_watts

Returns: power_watts (int): number of watts (e.g. `94`)

Inherited Members

Resources total, allocated, name, manufacturer, claim(), freeup(), died(), purchased(), category

class GPU(Resources): ▸ View Source

GPU class inheriting from base Resources class without modifications

GPU(name, manufacturer, total, allocated) ▸ View Source

Args: name (str): user-friendly name of resource instance (e.g. `Intel Core i9-9900K`) manufacturer (str): resource instance manufacturer (e.g. `Nvidia`) total (int): inventory total (how many are in the inventory pool) allocated (int): number allocated (how many are already in use), must be smaller than total

Inherited Members

Resources total, allocated, name, manufacturer, claim(), freeup(), died(), purchased(), category

class Storage(Resources): ▸ View Source

Intermediate Resources subclass with GB capacity

Storage(name, manufacturer, total, allocated, capacity_GB) ▸ View Source

Args: name (str): user-friendly name of resource instance (e.g. `Intel Core i9-9900K`) manufacturer (str): resource instance manufacturer (e.g. `Nvidia`) total (int): inventory total (how many are in the inventory pool) allocated (int): number allocated (how many are already in use), must be smaller than total capacity_GB (int): storage capacity in gigabytes (e.g. `120`)

capacity_GB

Returns: capacity_GB (int): storage capacity in gigabytes (e.g. `120`)

Inherited Members

Resources total, allocated, name, manufacturer, claim(), freeup(), died(), purchased(), category

class HDD(Storage): ▸ View Source

Storage subclass with size and rpm

HDD(name, manufacturer, total, allocated, capacity_GB, size, rpm) ▸ View Source

Args: name (str): user-friendly name of resource instance (e.g. `Intel Core i9-9900K`) manufacturer (str): resource instance manufacturer (e.g. `Nvidia`) total (int): inventory total (how many are in the inventory pool) allocated (int): number allocated (how many are already in use), must be smaller than total capacity_GB (int): storage capacity in gigabytes (e.g. `120`) size (str): storage size (e.g. `2.5"`) rpm (int): HDD's rpm count (e.g. `7000`)

size

Returns: size (str): storage size (e.g. `2.5"`)

rpm

Returns: rpm (int): HDD's rpm count (e.g. `7000`)

Inherited Members

Storage capacity_GB
Resources total, allocated, name, manufacturer, claim(), freeup(), died(), purchased(), category

class SSD(Storage): ▸ View Source

Storage subclass with interface

SSD(name, manufacturer, total, allocated, capacity_GB, interface) ▸ View Source

Args: name (str): user-friendly name of resource instance (e.g. `Intel Core i9-9900K`) manufacturer (str): resource instance manufacturer (e.g. `Nvidia`) total (int): inventory total (how many are in the inventory pool) allocated (int): number allocated (how many are already in use), must be smaller than total capacity_GB (int): storage capacity in gigabytes (e.g. `120`) interface (str): SSD's interface (e.g. `PCIe NVMe 3.0 x4`)

interface

Returns: interface (str): SSD's interface (e.g. `PCIe NVMe 3.0 x4`)

Inherited Members

Storage capacity_GB

Resources total, allocated, name, manufacturer, claim(), freeup(), died(), purchased(), category