

TreeInfo Class Reference

Public Member Functions

def [__init__](#) (self, [dlfile](#))

def [get_adk_version](#) (self)
Retrieve ADK version. [More...](#)

def [get_os_version](#) (self)
Retrieve OS version. [More...](#)

def [has_os](#) (self)
Check if contains an OS. [More...](#)

def [has_blacklists](#) (self)
Check if contains blacklists. [More...](#)

def [get_blacklists](#) (self)
Returns dictionary such as dict[device] = set(blacklisted bundles,...) [More...](#)

def [get_full_device_list](#) (self)
Return sorted list of devices. [More...](#)

def [get_bundle_device_list](#) (self, bundle)
Returns sorted list of all products on which this bundle can be installed. [More...](#)

Data Fields

[dlfile](#)[blacklists](#)[full_device_list](#)

Constructor & Destructor Documentation

[?__init__\(\)](#)

```
def __init__ ( self,
```

```
        dlfile
    )
```

Member Function Documentation

[**? get_adk_version\(\)**](#)

```
def get_adk_version ( self )
```

Retrieve ADK version.

Returns

adk version string or empty string if not found

[**? get_blacklists\(\)**](#)

```
def get_blacklists ( self )
```

Returns dictionary such as dict[device] = set(blacklisted bundles,...)

Returns

return blacklists dictionary

[**? get_bundle_device_list\(\)**](#)

```
def get_bundle_device_list ( self,
                             bundle
                             )
```

Returns sorted list of all products on which this bundle can be installed.

This takes into account blacklisting and targeting :param bundle: bundle node

Returns

sorted list of supported devices for this bundle

[**? get_full_device_list\(\)**](#)

```
def get_full_device_list ( self )
```

Return sorted list of devices.

Dlfile must contain an OS. Looks for devices in blacklists, then by llayer bundle names

Returns
sorted list of devices

[?](#) **get_os_version()**

```
def get_os_version ( self )
```

Retrieve OS version.

Returns
adk version string or empty string if not found

[?](#) **has_blacklists()**

```
def has_blacklists ( self )
```

Check if contains blacklists.

Returns
True if contains blacklists, False if not

[?](#) **has_os()**

```
def has_os ( self )
```

Check if contains an OS.

Returns
True if contains OS, False if not

Field Documentation

[?](#) **blacklists**

blacklists

? dlfile

dlfile

? full_device_list

full_device_list

The documentation for this class was generated from the following file:

- sys/import/packman/packmanlib/[treeinfo.py](#)