

# جافا في الاعتماديات تحليل

التبغعيات تجمعي يتم حيث، [ ] لصفوف التبعية مخطط لولادة المقدم البرمجي النص سأعدل استفسارك، مع للتعداد [ ] مثل أعلى المسوى في الحزم تجمعي يتم level=1، كان إذا المثال، سبيل على الحزم. هرمية من محمد مسوى حسب تحليل قبل أن يعني وهذا . [ ] مثل الثاني المسوى في تجمعيها يتم level=2، كان وإذا، [ ] هذه بين التبعيات في المخطط وسينظر الحزم، اسماء من أجزاء level أول على بناءً لمجموعات في الصفوف نجمع التبعيات الفردية. الصفوف من بدلًا لمجموعات

متطلباتك. تلبية وكيفية التغييرات شرح يليه المعدل، البرمجي النص أدناه

## المعدل البرمجي النص

```
import os
import sys
import re
from collections import defaultdict

def get_package(file_path):
    """
    .java.

Args:
    file_path (str): .java.

Returns:
    str: None
    .
    .
try:
    with open(file_path, 'r', encoding='utf-8') as f:
        for line in f:
            match = re.search(r'^\s*package\s+([\w.]+);', line)
            if match:
                return match.group(1)
except Exception as e:
    print(f"  : {file_path}: {e}")
return None

def get_specific_imports(file_path):
    """
    .java
```

Args:

```
file_path (str):      .java.

Returns:
list:      .

"""
imports = []
try:
    with open(file_path, 'r', encoding='utf-8') as f:
        for line in f:
            match = re.search(r'^\s*import\s+([\w.]+);', line)
            if match:
                imp = match.group(1)
                #                 )      java.util.*;)

                if not imp.endswith('.*'):
                    imports.append(imp)
except Exception as e:
    print(f"  : {file_path}: {e}")
return imports
```

def get\_package\_group(full\_class\_name, level):

```
"""
'level'      .

```

Args:

```
full_class_name (str):      ) "org.springframework.boot.App").
level (int):      ) 1 "org" 2 "org.springframework".
```

Returns:

```
str:      ) "org"      "org.springframework").
```

```
"""
package = '.'.join(full_class_name.split('.')[ :-1])  #
parts = package.split('.')
if len(parts) <= level:
    return package  #
else:
    return '.'.join(parts[:level])  #      'level'


```

```
if __name__ == '__main__':
#      :      root_directory  level
```

```

if len(sys.argv) != 3:
    print(" python script.py <root_directory> <level>")
    sys.exit(1)

root_dir = sys.argv[1]
try:
    level = int(sys.argv[2])
    if level < 1:
        raise ValueError
except ValueError:
    print(" (")
    sys.exit(1)

all_classes = set()

# :
for root, dirs, files in os.walk(root_dir):
    for file in files:
        if file.endswith('.java'):
            file_path = os.path.join(root, file)
            package = get_package(file_path)
            if package:
                class_name = file.replace('.java', '')
                full_class_name = f"{package}.{class_name}"
                all_classes.add(full_class_name)

# :
group_dependencies = set()

for root, dirs, files in os.walk(root_dir):
    for file in files:
        if file.endswith('.java'):
            file_path = os.path.join(root, file)
            package = get_package(file_path)
            if package:
                class_name = file.replace('.java', '')
                full_class_name = f"{package}.{class_name}"
                importer_group = get_package_group(full_class_name, level)
                imports = get_specific_imports(file_path)

```

```
for imp in imports:
    #
    #
    if imp in all_classes and imp != full_class_name:
        imported_group = get_package_group(imp, level)
        if imported_group != importer_group:
            group_dependencies.add((importer_group, imported_group))

#
#           DOT
print('digraph G {')
for from_group, to_group in sorted(group_dependencies):
    print(f'    "{from_group}" -> "{to_group}";')
print('}')  
}
```

الغیرات شرح

- ## 1. ال أوامر ال سطر م ت غ ي رات .

الآن: `python script.py <root_directory>`

الى الامثلية: python script.py <root\_directory> <level>

وجود دن ال برمجي ال نص يتحقق ال حزم. هرميّة مسّتوى يحدّد ال ذي، level ال ثانوي، ال أوامر ال سطريّة متغيّر دعم إضافيّة تم موجّب. صحيح عدد هو level أن من وتأكد بالضبط ممتغّرین

- ## 2. دالة get\_package\_group: جديدة

المحدد. level علی بن ائل صف الـ حـ زـ مـة مـجـمـوـعـة لـ حـ سـ اـبـ دـالـة إـضـافـة تـم

الحزمة يس تخرج، مثل الـكامل الـصـفـلـاـسـمـ بـالـنـسـبـةـ

اجزاء: level اول ويأخذ، level اول ويقسم،

لـ `level=1`: دعـ اـنـ اـذـ.

الكمالة الاحزمة يعيده،  $\text{level}=3$  مع `level=3` مثل `level=1`.  
أجزاء عدد من أقل لديها احزمة كانت إذا `level=1`.

- ### 3. الاتبعیات تجمیع:

الفرديه. الاصفوف بین الالتباعیات لتخزین (set) استخدمن defauldict (set) :الاصلیة

كزواج الـ `group_dependencies` (from\_group, to\_group).

ڪل: فص

||**importer\_group**|| مجموعتھی حسب باسٹ خدام get\_package\_group.

imp != full\_class\_name: نفسه الصرف يكون ولو imp in all\_classes: المشرع داخل محددasti رادل كل

[[ imported\_group ]].

إلى حافة يضيّف، `imported_group != importer_group` مختلفة المجموعات كانت إذا `group_dependencies`.

واحدة. حافة إلى المجموعات نفس بين المتعددة التبعيات تؤدي لذلك الفردي، `set` يضم من

#### 4. خروج:

مثـالـاـ لـفـرـديـاـ الصـفـوفـ بـيـنـ حـوـافـ طـبـاعـهـ:ـ الـأـصـلـيـهـ

ـ.

ـ مـثـالـاـ لـحـزـمـ مـجـمـوـعـاتـ بـيـنـ حـوـافـ طـبـاعـهـ:ـ الـمـعـدـلـهـ

ـ.

#### متطلباتك تلبية كيفية

التبعيات. تحليل قبل الحزم لتجمـع `level` متـغـيرـ الـبـرـمـجـيـ الـنـصـ يـقـبـلـ الـآنـ:ـ الـمـسـتـوـيـاتـ دـعـمـ

ـ الـمـثـالـ،ـ سـبـيـلـ عـلـىـ .ـ مـثـالـ الـأـعـلـىـ الـحـزـمـ حـسـبـ الصـفـوفـ جـمـيـعـ تـجـمـيـعـ:ـ 1ـ الـمـسـتـوـيـ

ـ وـ

ـ دـاـخـلـ بـيـنـ هـمـ اـسـتـيـرـادـاتـ تـظـهـرـ لـاـ لـذـلـكـ ،ـ 2ـ مـجـمـوـعـةـ فـيـ كـلـ اـهـمـاـ

ـ وـ

ـ كـحـوـافـ.

ـ سـبـيـلـ عـلـىـ .ـ مـثـالـ الـحـزـمـ مـنـ مـسـتـوـيـيـنـ أـوـلـ حـسـبـ الصـفـوفـ تـجـمـيـعـ:ـ 2ـ الـمـسـتـوـيـ

ـ حـافـةـ يـخـلـقـ

ـ إـلـىـ

ـ مـنـ اـسـتـيـرـادـ الـمـثـالـ

ـ إـلـىـ

ـ مـنـ

ـ تـحـلـيـلـ قـبـلـ `level` عـلـىـ بـنـاءـ صـفـ لـكـلـ الـحـزـمـ مـجـمـوـعـةـ الـبـرـمـجـيـ الـنـصـ يـحـدـدـ:ـ الـتـبـعـيـاتـ تـحـلـيـلـ قـبـلـ الـحـزـمـ تـجـمـيـعـ

ـ الـفـرـديـهـ

ـ الـصـفـوفـ وـلـيـسـ الـمـجـمـوـعـاتـ

ـ بـيـنـ الـتـبـعـيـاتـ أـنـ يـضـمـنـ هـمـ اـسـتـيـرـادـهـ

ـ الـمـثـالـ:ـ لـحـافـةـ بـالـنـسـبـةـ:ـ الـتـوـافـقـ مـثـالـ

ـ وـاحـدـةـ .ـ مـجـمـوـعـةـ حـافـةـ إـضـافـةـ يـتـمـ لـاـ لـذـلـكـ ،ـ 1ـ فـيـ كـلـ اـهـمـاـ

ـ 2ـ مـعـ

ـ

ـ 3ـ مـعـ

ـ

ـ الـمـخـتـلـفـةـ.ـ الـمـجـمـوـعـاتـ هـذـهـ بـيـنـ حـافـةـ إـضـافـةـ

#### الاستخدام أمثلة

ـ إـلـاـمـرـ ـ:ـ py~thon~ s~c~ri~pt~.~py~ /~p~a~t~h~/~t~o~/~p~r~o~j~e~c~t~ ~1~

ـ إـلـخـ.ـ ،ـ 1ـ الـمـجـمـوـعـاتـ

ـ حـزـمـةـ فـيـ صـفـ ـ حـزـمـةـ فـيـ صـفـ اـسـتـورـدـ إـذـاـ "org" ~-> "com"ـ الـخـرـجـ:ـ يـتـضـمـنـ قـدـ

ـ إـلـاـمـرـ ـ:ـ py~thon~ s~c~ri~pt~.~py~ /~p~a~t~h~/~t~o~/~p~r~o~j~e~c~t~ ~2~

ـ إـلـخـ.ـ ،ـ 2ـ الـمـجـمـوـعـاتـ

ـ "org.springframework" ~-> "org.apache".ـ الـخـرـجـ:ـ يـتـضـمـنـ قـدـ

#### ملحوظات

ـ اـسـتـيـرـادـاتـ فـقـطـ الـمـشـرـوعـ دـاـخـلـ الـتـبـعـيـاتـ تـضـمـيـنـ مـنـ الـأـصـلـيـ سـلـوكـهـ عـلـىـ الـبـرـمـجـيـ الـنـصـ يـظـلـ

ـ

ـ الـذـاتـيـهـ.ـ وـالـتـبـعـيـاتـ الـجـمـلـهـ اـسـتـيـرـادـاتـ وـاـسـتـبـعـادـ

ـ الـأـصـلـيـ.ـ الـبـرـمـجـيـ الـنـصـ مـعـ موـافـقـاـ حـزـمـةـ،ـ إـعـلـانـ بـدـوـنـ الـصـفـوفـ تـجـاهـلـ يـتـمـ

مثلاً أدوات باستخدام مشاهدته يمكن، صيغة في موجه مخططه هو الخرج

الحزم. هرمية مستوى حسب الاتباعيات لتجمیع طلبك الامunal البدرومجي الانص هذا يرضي أن يجب