

산업인공지능학과

# Mini Project

---

산업인공지능개론

2020254003

원형일

## Contents

I. 대상 설명

II. Cording

III. 결과

# I. 대상설명 (1)

## 산업인공지능개론



조립불량

조립과정상의 불량

자재불량

납품된 자재 불량.

제작불량

본딩등 제작상의 문제

포장불량

조립이후 출하전의 문제

키워드

도어, 재단면, 접착면,본딩,포장박스,완충재,주소

## II. Cording(1)

## 산업인공지능개론

```
1 !pip install durable_rules
2
3 from durable.lang import *
4
5 with ruleset('팻도어'):
6     @when_all(c.first << (m.predicate=='팻도어의') & (m.object == '품질검사법입니다'),
7                 (m.predicate=='제품의상태중') & (m.object=='불량의경우는') & (m.subject==c.first.subject))
8     def 재단불량(c):
9         c.assert_fact({'subject':c.first.subject, 'precate': '재단면이', 'object': '고르지않을 경우'})
10
11     @when_all((m.predicate == '제품의상태중') & (m.object=='불량의경우는'))
12     def 재단불량(c):
13         c.assert_fact({'subject': c.m.subject, 'precate':'재단면이','object':'거칠 경우'})
14
15     @when_all(c.first << (m.predicate=='팻도어의') & (m.object=='품질검사법입니다'),
16                 (m.predicate=='조립상태중 ') & (m.object=='불량의경우는') & (m.subject==c.first.subject))
17     def 도어불량(c):
18         c.assert_fact({'subject': c.first.subject, 'preicate': '도어가', 'object':'부드럽지 않을 경우' })
19
20     @when_all((m.predicate == '제품의상태중') & (m.object=='불량의경우는'))
21     def 접착불량(c):
22         c.assert_fact({'subject': c.m.subject, 'precate':'접착부분이','object':'일어날 경우'})
23
24     def 접착불량(c):
25         c.assert_fact({'subject': c.m.subject, 'precate':'마감재에','object':'주름이 있을 경우'})
26
27     @when_all((m.predicate== '조립상태중 ') & (m.object=='불량의경우는'))
28     def 형태불량(c):
29         c.assert_fact({'subject': c.m.subject, 'predicate': '재단이', 'object':'일정하지 않을 경우'})
30
31     @when_all((m.predicate == '자재의상태중 ') & (m.object == '불량인경우는'))
32     def 잠금불량(c):
33         c.assert_fact({'subject': c.m.subject, 'predicate': '도어가', 'object': '작동되지 않을 경우'})
34
35     @when_all((m.precate == "조립상태중") & (m.object == '불량인경우는'))
36     def 본딩불량(c):
37         c.assert_fact({'subject':c.m.subject, 'predicate': '본딩이 ', 'object': '지저분할 경우'})
38
```



## II. Cording(2)

## 산업인공지능개론

```
39 @when_all((m.precate == "출하준비중") & (m.object == '불량인 경우'))
40 def 포장불량(c):
41     c.assert_fact({'subject':c.m.subject, 'predicate': '포장박스가 ', 'object': '찌그러지거나 파손될 경우'})
42
43 @when_all((m.precate == "출하준비중") & (m.object == '불량인 경우'))
44 def 포장불량(c):
45     c.assert_fact({'subject':c.m.subject, 'predicate': '완충재가 ', 'object': '부족할 경우'})
46
47 @when_all((m.precate == "출하준비중") & (m.object == '불량인 경우'))
48 def 포장불량(c):
49     c.assert_fact({'subject':c.m.subject, 'predicate': '포장박스가 ', 'object': '태이핑불량일 경우'})
50
51 @when_all((m.precate == "출하준비중") & (m.object == '불량인 경우'))
52 def 포장불량(c):
53     c.assert_fact({'subject':c.m.subject, 'predicate': '주소가입이 ', 'object': '누락될 경우'})
54
55 def 포장불량(c):
56     c.assert_fact({'subject':c.m.subject, 'predicate': '주소가입이 ', 'object': '오기입될 경우'})
57
58 @when_all(+m.subject)
59 def outfut(c):
60     print('Fact: {0} {1} {2}'.format(c.m.subject, c.m.predicate, c.m.object))
61
62 assert_fact('됐도어',{'subject':'조립불량', 'predicate': '재단면이', 'object': '고르지않을 경우'})
63 assert_fact('됐도어',{'subject':'조립불량', 'predicate': '재단면이', 'object': '거칠 경우'})
64 assert_fact('됐도어',{'subject':'조립불량', 'predicate': '마감재에', 'object': '주름이 있을 경우'})
65 assert_fact('됐도어',{'subject':'조립불량', 'predicate': '도어가', 'object': '부드럽지 않을 경우'})
66 assert_fact('됐도어',{'subject':'조립불량', 'predicate': '도어가', 'object': '작동하지 않을 경우'})
67 assert_fact('됐도어',{'subject':'조립불량', 'predicate': '전착면이', 'object': '일어날 경우'})
68 assert_fact('됐도어',{'subject':'자재불량', 'predicate': '재단이', 'object': '일정하지 않을 경우'})
69 assert_fact('됐도어',{'subject':'자재불량', 'predicate': '도어가', 'object': '작동하지 않을 경우'})
70 assert_fact('됐도어',{'subject':'제작불량', 'predicate': '분당이', 'object': '지저분할 경우'})
71 assert_fact('됐도어',{'subject':'포장불량', 'predicate': '포장박스가', 'object': '찌그러지거나 파손될 경우'})
72 assert_fact('됐도어',{'subject':'포장불량', 'predicate': '포장박스가', 'object': '태이핑불량일 경우'})
73 assert_fact('됐도어',{'subject':'포장불량', 'predicate': '완충재가', 'object': '부족할 경우'})
74 assert_fact('됐도어',{'subject':'포장불량', 'predicate': '주소가', 'object': '누락될 경우'})
75 assert_fact('됐도어',{'subject':'포장불량', 'predicate': '주소가', 'object': '오기입될 경우'})
76 assert_fact('됐도어',{'subject':'조립불량', 'predicate': '분당이', 'object': '지저분할 경우'})# Welcome to
```



Collecting durable\_rules

Downloading [https://files.pythonhosted.org/packages/91/f2/3194b4eaf7260bae0a4f046dfdda5db2ed9d58b3d7a051e375b3c14d8f26/durable\\_rules-2.0.28.tar.gz](https://files.pythonhosted.org/packages/91/f2/3194b4eaf7260bae0a4f046dfdda5db2ed9d58b3d7a051e375b3c14d8f26/durable_rules-2.0.28.tar.gz) (57kB)

61kB 3.2MB/s

Building wheels for collected packages: durable-rules

Building wheel for durable-rules (setup.py) ... done

Created wheel for durable-rules: filename=durable\_rules-2.0.28-cp37-cp37m-linux\_x86\_64.whl size=176354 sha256=b43339ccd85b4b665f9080e1a656a598b50a2bf5a4f192e8d0cf177153b72d5b

Stored in directory: /root/.cache/pip/wheels/97/f5/6b/dabbc5d4c2571374b0e0c49d8c80c449220d3061c20b057ba4

Successfully built durable-rules

Installing collected packages: durable-rules

Successfully installed durable-rules-2.0.28

Fact: 조립불량 재단면이 고르지않을 경우

Fact: 조립불량 재단면이 거칠 경우

Fact: 조립불량 마감재에 주름이 있을경우

Fact: 조립불량 도어가 부드럽지 않을 경우

Fact: 조립불량 도어가 작동하지 않을 경우

Fact: 조립불량 접착면이 일어날 경우

Fact: 자재불량 재단이 일정하지 않을 경우

Fact: 자재불량 도어가 작동하지 않을 경우

Fact: 제작불량 본딩이 지저분할 경우

Fact: 포장불량 포장박스가 찌그러지거나 파손될경우

Fact: 포장불량 포장박스가 테이핑불량일 경우

Fact: 포장불량 완충재가 부족할경우

Fact: 포장불량 주소가 누락될경우

Fact: 포장불량 주소가 오기입될경우

Fact: 조립불량 본딩이 지저분할 경우

{'\$s': 1, 'id': 'sid-0', 'sid': '0'}