

# CAPSTONE DESIGN

냉장고를 봇탁해!

2조

이시현, 손병민, 범종원, 조정은, 김유나

# 목차

## 현재 프로젝트 진행 상황

- 객체탐지 알고리즘 재구성
- 3D 프린터
- DB 연동

## 다음주 프로젝트 진행 예정 사항

# 3D 모델링

## IT융합대학 3D프린터 출력 신청 안내

### ▶ 신청 방법

이메일([it@chosun.ac.kr](mailto:it@chosun.ac.kr))로 설계도 파일([영문명.stl](#))과 함께 신청

※ 추가 작성내용

- 1) 학생 개인정보 「학과·학번·성명·연락처」
- 2) 출력목적 및 교과목명·담당교수 \*사적인 목적으로 출력 불가
- 3) 소요 필라멘트(ABS-A100 / PLA+) \*미입력 시 자율적으로 출력합니다.
  - M.space 3D프린터실 보유 프린터: PLA+ 3대, ABS-A100 2대 \*필라멘트 교체 시 노즐 막힘
  - ABS-A100: 후처리·가공이 용이하지만 수축으로 인한 하단 들뜸현상 발생 심함
  - PLA+: 후처리·가공이 다소 힘들지만 수축에 있어서 ABS에 비해 안정적

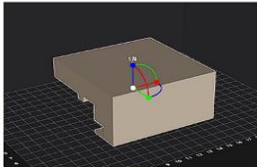
### ▶ 소요 시간

신청 후 2~3일 소요(주말 제외), 출력 완료 후 연락 안내

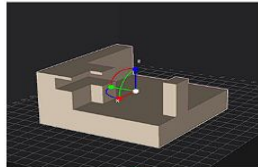
### ▶ 설계 최대 규격 \*주의사항 ②번 참조

- 가로(x축) - 23cm 미만
- 세로(y축) - 18cm 미만
- 높이(z축) - 19cm 미만

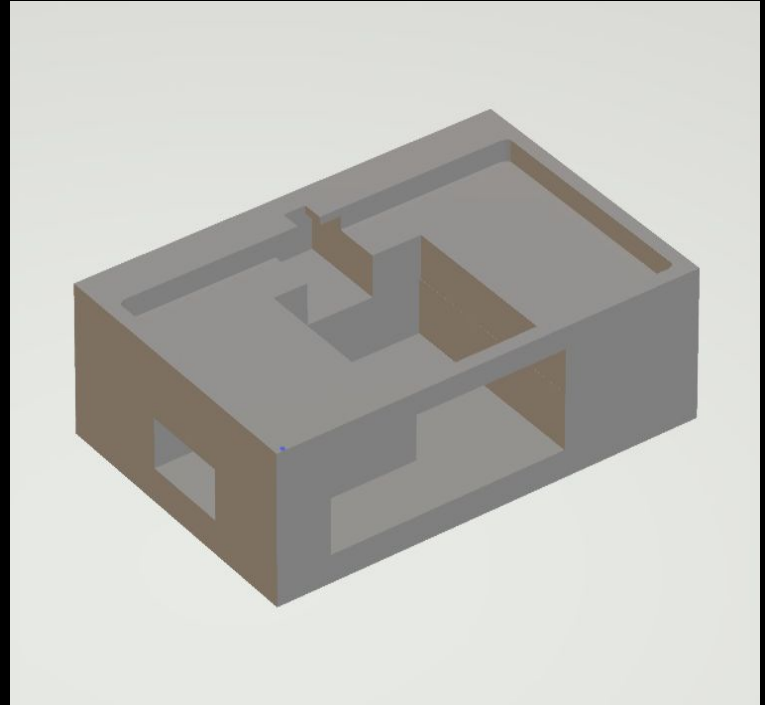
※ 다음과 같이, 최대한 바닥에 닿는 면이 넓도록 높여서 제출 바랍니다.



X



O



## 문제 사항 - 하드웨어

사용자가 버튼을 사용하지 않는 물체 인식 자동화 알고리즘

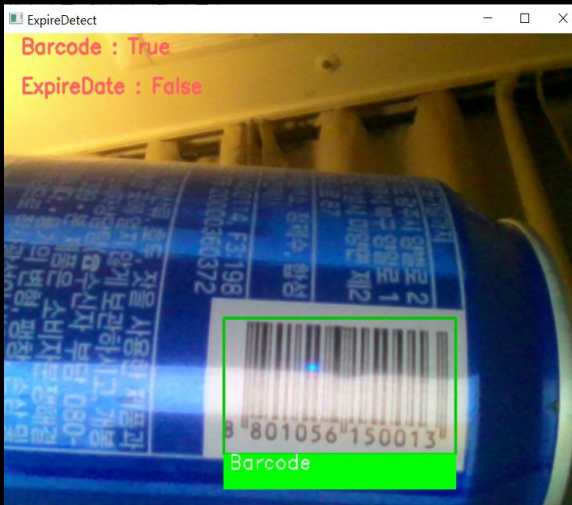
-> 좌표값 저장후 앞뒤 평균값으로 오차 계산

라즈베리파이 모듈 설치에 많은 시간 소요 -> 오류

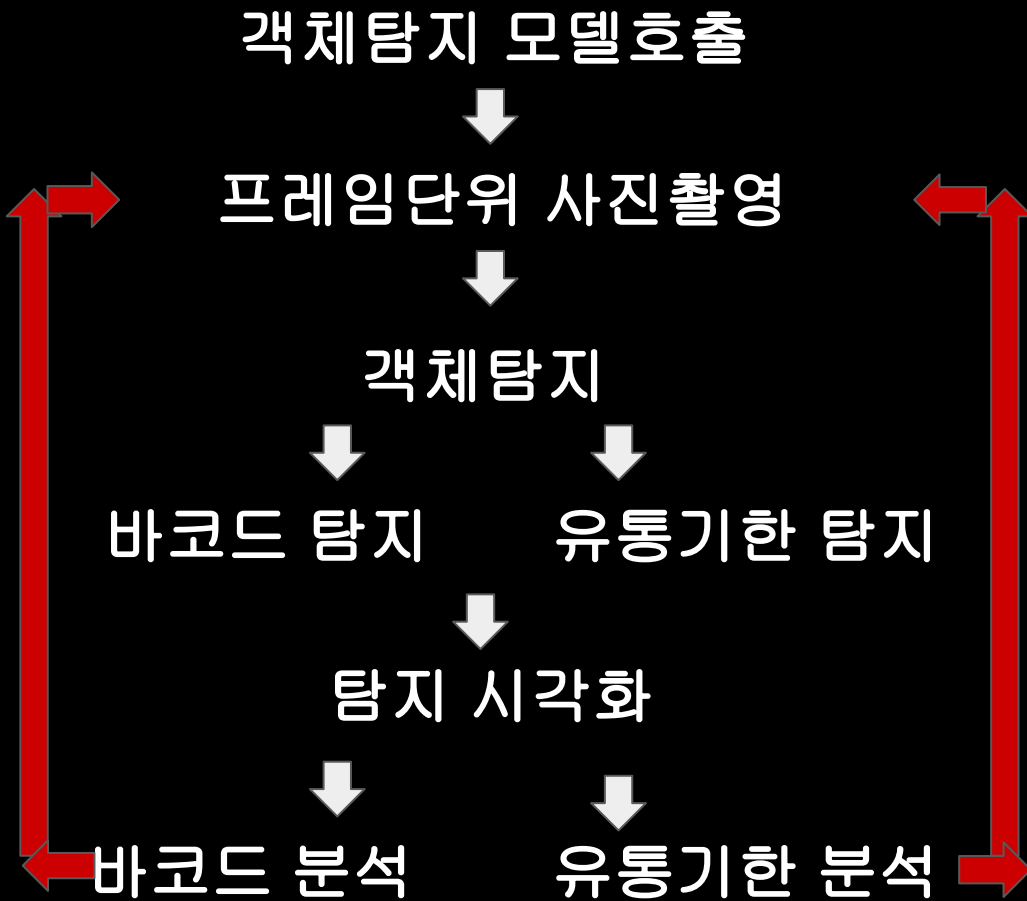
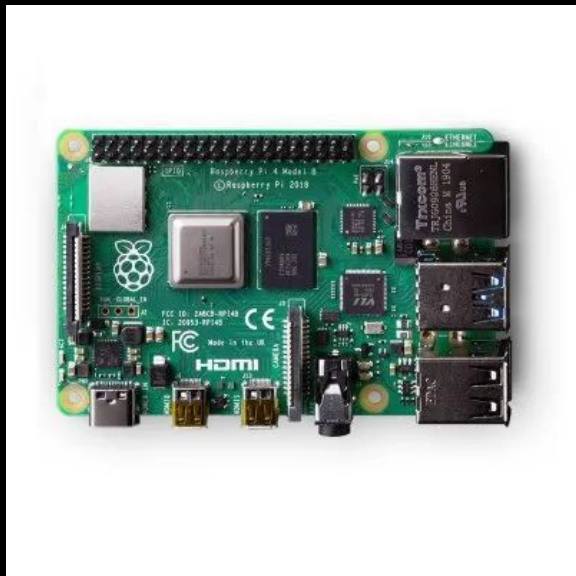
바코드 이미지 데이터 부족

# 탐색 알고리즘 재구성

Barcode : False  
ExpireDate : False



# 하드웨어 고려사항



# 이번주 진행 상황



챗봇 직접 입력 기능 DB 연동

바코드, 이미지 학습 데이터 연결 및 DB 연동



하드웨어 추가 구현

# 이전 DB 진행 상황

```
chatbot / application.py
application.py models.py app.py

48     }
49     }
50     }
51     }
52     }
53     }
54     # 답변 전송
55     return jsonify(res)
56
57 * sql = """INSERT INTO items(item_type, item_name, item_code, expirt_date, item_number
58     VALUES('유형', '이름', '코드', '유통기한', '숫자');"""
59
60 cursor.execute(sql)
61 db.commit()
62 result = cursor.fetchall()
63 # Database 닫기
64 db.close()
65
66 if __name__ == '__main__':
67     # application.run(host='9.36.111.61', port=54981, threaded=True)

디버그 터미널 검색 리소스 모니터 힌트

lt
result.read()
File "/usr/local/lib/python3.7/site-packages/pymysql/connections.py", line 1157, in read
    first_packet = self.connection._read_packet()
File "/usr/local/lib/python3.7/site-packages/pymysql/connections.py", line 729, in _read_packet
    packet.raise_for_error()
File "/usr/local/lib/python3.7/site-packages/pymysql/protocol.py", line 221, in raise_for_error
    err.raise_mysql_exception(self._data)
File "/usr/local/lib/python3.7/site-packages/pymysql/err.py", line 143, in raise_mysql_exception
    raise errorclass(errno, errval)
pymysql.err.ProgrammingError: (1064, "You have an error in your SQL syntax; check the manual that c
responds to your MySQL server version for the right syntax to use near ')','code','date','number')'
tline 2")
root@goorm:/workspace/chatbot# python application.py
root@goorm:/workspace/chatbot# python application.py
root@goorm:/workspace/chatbot#
```

DBeaver 23.0.0 - items

Properties Data 엔티티 관계도 3.34.133.115 Databases chatbot Tables items

items Enter a SQL expression to filter results (use Ctrl+Space)

	rec item_type	rec item_name	rec item_code	rec expirt_date	rec item_number
1	값	값	값	값	값
2	값	값	값	값	값
3	값	값	값	값	값
4	값	값	값	값	값
5	값	값	값	값	값
6	type	name	code	date	number
7	type	name	code	date	number

Record

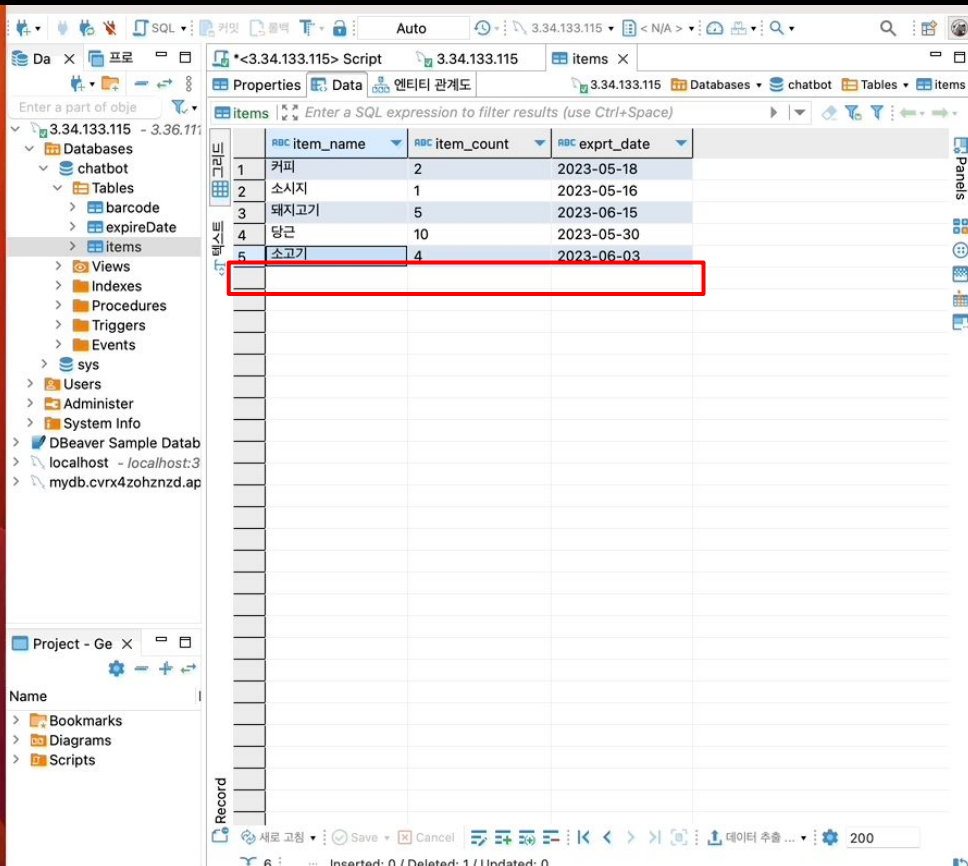
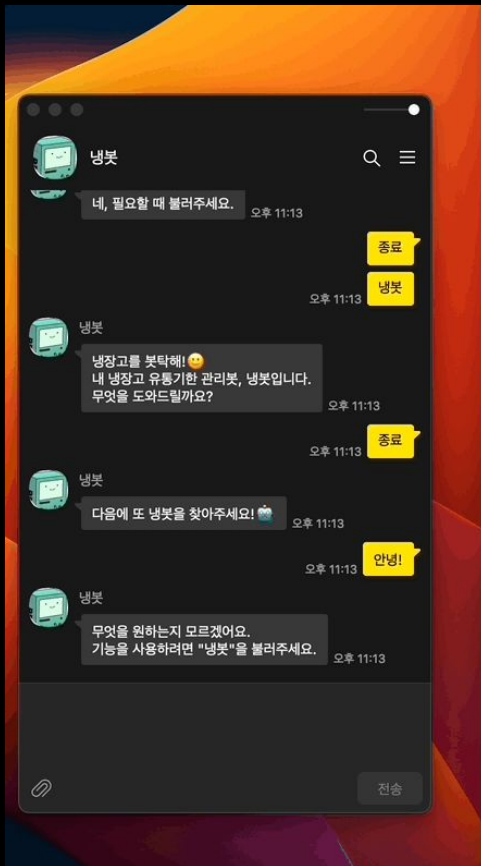
새로 고침 Save Cancel 200 7

8 row(s) fetched - 14ms, on 2023-05-09 at 23:59:15

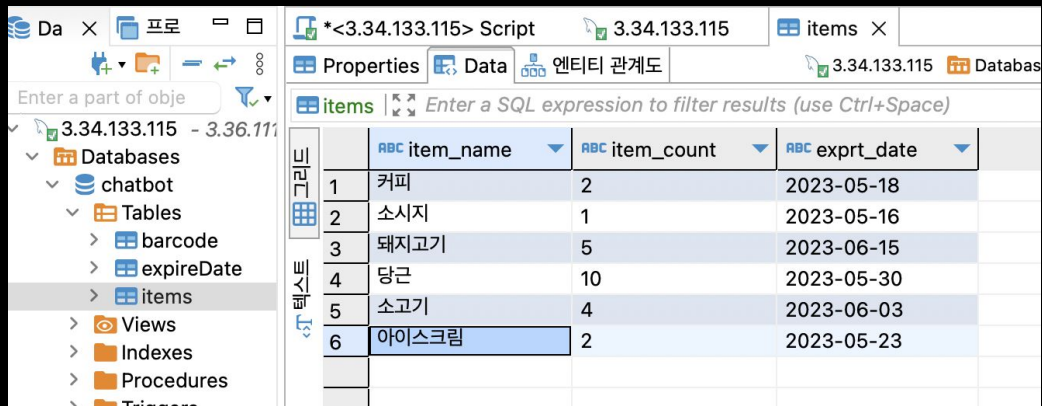
KST kn KR



# 현재 DB 진행 상황



# 현재 DB 진행 상황



The screenshot shows a database management interface. On the left, a tree view displays the database structure: '3.34.133.115' - '3.36.111' containing 'Databases' (chatbot), 'Tables' (barcode, expireDate, items, Views, Indexes, Procedures, Triggers). The 'items' table is selected. The main area shows the table's properties and data. The 'Data' tab is active, displaying a table with columns 'item\_name', 'item\_count', and 'expirt\_date'. The data is as follows:

	ABC item_name	ABC item_count	ABC expirt_date
1	커피	2	2023-05-18
2	소시지	1	2023-05-16
3	돼지고기	5	2023-06-15
4	당근	10	2023-05-30
5	소고기	4	2023-06-03
6	아이스크림	2	2023-05-23

```
^Croot@ggoorm:/workspace/chatbot# python models.py
냉장고 둘러보기
('커피', '2', '2023-05-18')
('소시지', '1', '2023-05-16')
('돼지고기', '5', '2023-06-15')
('당근', '10', '2023-05-30')
('소고기', '4', '2023-06-03')
('아이스크림', '2', '2023-05-23')
* Serving Flask app "models"
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:54832
* Running on http://172.17.0.22:54832
Press CTRL+C to quit
* Restarting with stat
```

# 다음주 프로젝트 진행 계획

냉장고 둘러보기(DB 보여주기) 기능 구현

유통기한 가까워질수록 강조 표시

바코드, 이미지 학습 데이터 연결 및 DB 연동

라즈베리파이 코드 구현

3D 프린터 조립