



**식단의 모든 것, 다이어트마이트**

# 목차

---

## 프로젝트 개요

리뷰  
기획 배경  
기획 목적  
개발 일정

01

## 개발 환경

개발언어  
개발툴  
원격협업

02

## 설계

구조 설계  
DB 설계  
화면 설계

03

## 구현

기본 설정  
주요 기능  
코드 구현

04

## 시연

현장 시연  
QnA  
참고사이트

05

# 프로젝트 개요

01 1차 프로젝트 리뷰

02 기획 배경

03 기획 목적

04 개발 일정

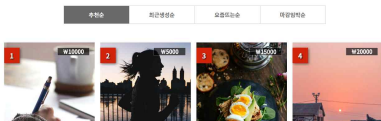
# 01 1차 프로젝트 리뷰- STAMPUS



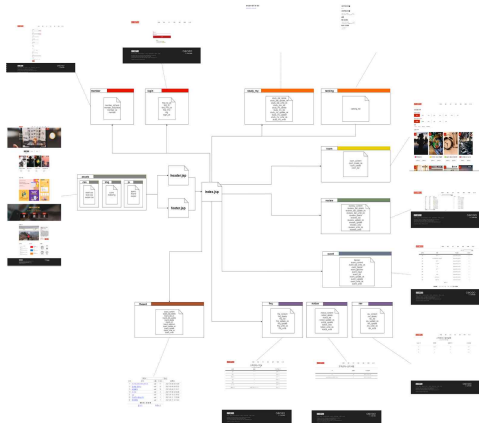
• 서비스 목적 : 인증 시스템을 갖춘 소셜 웹서비스

• 주요 기능

- ✓ 인증 방법 선택 → 방생성 및 참여
- ✓ 인증 자동 시스템 구축 → 인증 현황 파악
- ✓ 보증금 제도 도입 → 서비스의 차별화



# 01 1차 프로젝트 리뷰- STAMPUS



- 서비스 목적 : 인증 시스템을 갖춘 소셜 웹서비스
- 주요 기능
  - ✓ 인증 방법 선택 → 방생성 및 참여
  - ✓ 인증 자동 시스템 구축 → 인증 현황 파악
  - ✓ 보증금 제도 도입 → 서비스의 차별화
- 구조
  - 각 페이지 → **CRUD** 구현한 게시판
  - 게시판 중심 : JSP/Servlet + DB구축/웹표준
    - ❖ FE/BE구축 및 웹 퍼블리싱

## 02 기획 배경



1. Front-end 프레임워크 활용 → 기능 구현 중심
2. 데이터 수집 → 크롤링 및 API 활용
3. 데이터 분석 → 시각화(python Lib), 뷰(Chart.js)
4. 데이터 예측 → ML 모델 선정 및 성능 개선
5. 데이터 활용 → MVC 모델 이용 웹애플리케이션 구현

**“머신러닝 기반 웹서비스”**

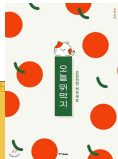
## 02 기획 배경

---

**“오늘 뭐 먹지?”**

## 02 기획 배경

오늘  
뭐먹지?  
릴리버리



“오늘 뭐 먹지?”



오늘 뭐 먹지?  
일상식 BEST

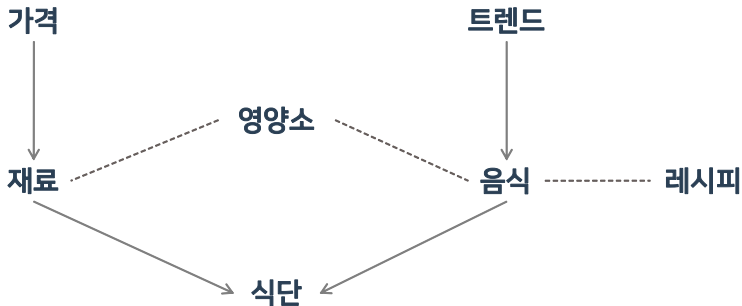




## 02 기획 배경

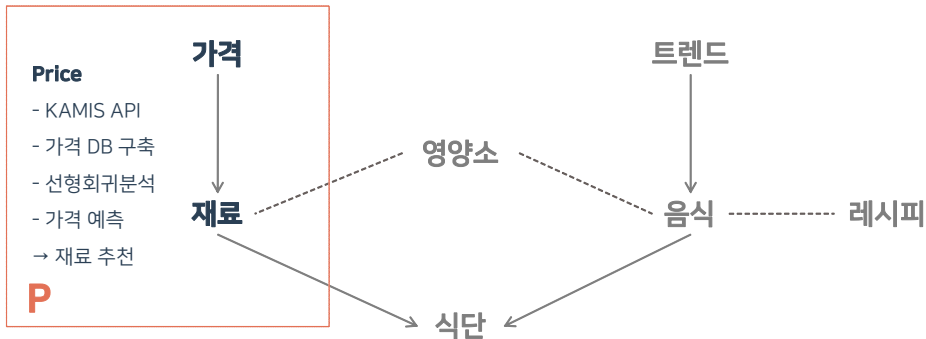


## 03 기획 목적

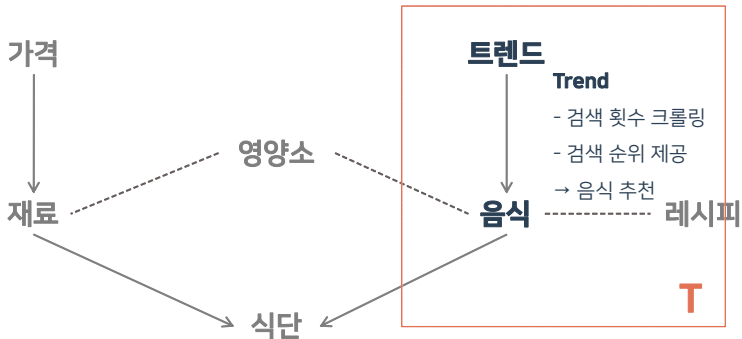


오늘 뭐먹을지 판단 도식화 → **5개 Sector** 로 분할

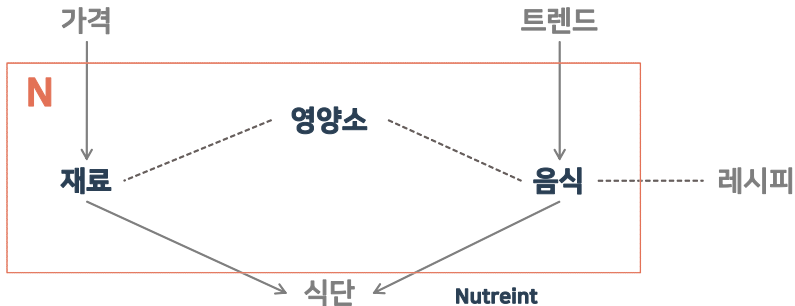
## 03 기획 목적



## 03 기획 목적

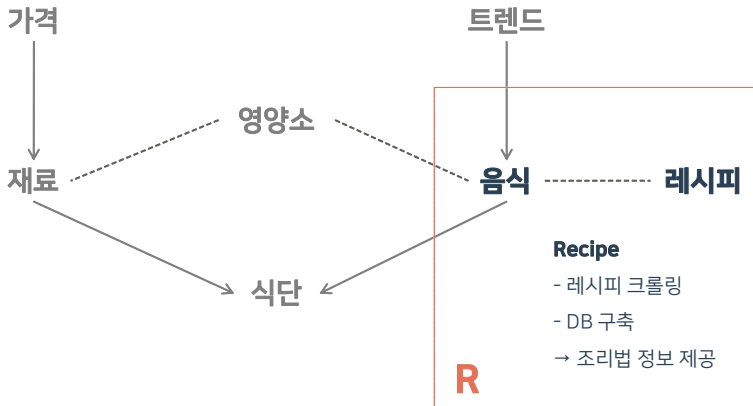


## 03 기획 목적

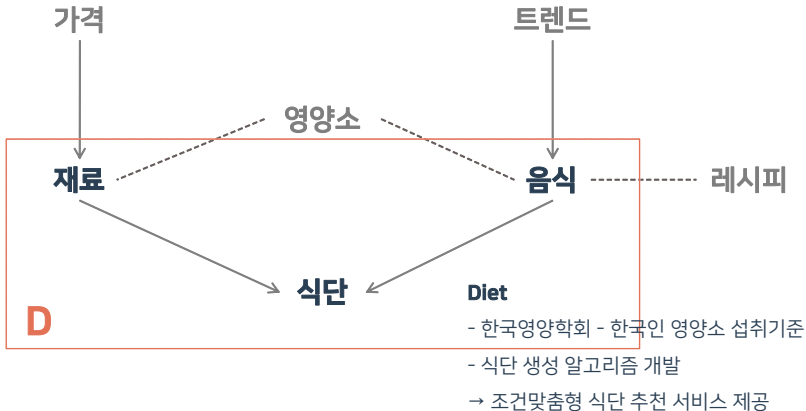


- 식약처 식품영양성분 데이터베이스
- 통합 영양성분 DB 구축
- 재료 및 음식별 성분 정보 제공

## 03 기획 목적



## 03 기획 목적



**“식재료 가격 및 트렌드 동향에 따른  
맞춤형 식단 추천 서비스”**



## 04 개발일정

01.22

### 1차 프로젝트 발표

1주일간 자체 팀휴가  
재충전 및 개인정비

02.08

### 분석 및 설계

업무 분할, DB 모델링,  
화면설계

02.22



### 서비스명 및 로고 선정

03.08

### 2차 개발

주요 기능 구현, 데이터  
시각화 작업, 링크 작업  
및 디버깅

03.17

### 2차 프로젝트 발표

시연 및 피드백

02.01

### 아이템 서칭 및 주제 선정

데이터 접근성, 데이터 분석 필요성,  
데이터 예측 실현 가능성, 데이터 활용  
및 서비스 구현 가능성

02.15

### 1차 개발

협업환경 구축, 프로세스 설계, 테이블  
정규화, HTML 구조 설계, 부트스트랩  
디자인 선정, 크롤링 및 API 신청, 데이  
터 분석

03.01

### 1차 테스트

기본화면 설계 완료, 테이블  
설계 완료, 개인 구현범위 설정,  
페이지 테스트

03.15

### 2차 테스트/배포

페이지 구현 완료, 버그 수정,  
추가기능 구현, 최종 기능  
테스트, Git 최종 Merge, 코드  
리뷰, 톡캣 서버 배포

## 개발 환경

01 개발 언어 및 툴

02 협업 - Git

03 협업 - Slack

# 01 개발 언어 및 툴

구분	1차 프로젝트
FE	HTML, CSS, JS, jQuery
BE	JSP/Servlet, Tomcat
DB	Oracle
ML	-
Tools	Git, Photoshop, Slack, eclipse

VS.

구분	2차 프로젝트
FE	+ Bootstrap, Chart.js
BE	+ Spring
DB	MariaDB, HeidiSQL
ML	+ Scikit-learn
Tools	+ PyCharm, JupyterNB, STS

**159 commits**  
159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 1040 1041 1042 1043 1044 1045 1046 1047 1048 1049 1050 1051 1052 1053 1054 1055 1056 1057 1058 1059 1060 1061 1062 1063 1064 1065 1066 1067 1068 1069 1070 1071 1072 1073 1074 1075 1076 1077 1078 1079 1080 1081 1082 1083 1084 1085 1086 1087 1088 1089 1090 1091 1092 1093 1094 1095 1096 1097 1098 1099 1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 1110 1111 1112 1113 1114 1115 1116 1117 1118 1119 1120 1121 1122 1123 1124 1125 1126 1127 1128 1129 1130 1131 1132 1133 1134 1135 1136 1137 113

jeeltwon · 9525b52 · 2 changed files · New

```
food@srcwm...$DietMakerSecond.java
food@srcwm...$webapp...$index.jsp

@@ -48,7 +48,7 @@ public class DietMakerSecond {
    }
    check_out(people.nut_lb, cur_nut, people.nut_ub, ddao);
    i += 1;
}
System.out.println(i+"번 try=====");
System.out.println(i+"번 try=====");
}
return meals;
}
}

@@ -56,7 +56,7 @@ public class DietMakerSecond {
    public static void out_meal() {
        System.out.println("=====오늘의 식단=====");
        for (FoodDto fdto: meals) {
            System.out.println(fdto.getFood_cate()+ " "+fdto.getFood_name());
            System.out.println("[*"+fdto.getFood_cate()+"] "+fdto.getFood_name());
        }
    }
}

@@ -77,7 +77,7 @@ public class DietMakerSecond {
    FoodDto food = get_food(i, ddao);
    //System.out.println("후보음식: "+ food.getFood_name());
    if(is_under_ub(nut_ub, food)) {
        System.out.println("영양성분 만족 ===>" +food.getFood_name()+food.getFood_cate());
        System.out.println("칼로리성분 만족 ===>["+food.getFood_cate()+"] "+food.getFood_name());
        add_food(food, ddao);
        return true;
    } else if(meals.size()==0) {
    }
}

@@ -119,7 +119,7 @@ public class DietMakerSecond {
    }
    if(cur_nut[i].getFoodnutIndex(fdto,i)>nut_ub[i]) {
        add_rejection(i);
    }
    System.out.println(nut_list[i]+"초과 ==> "+(cnt_rejection[i]+1)+"회 거절");
    System.out.println(nut_list[i]+"*초과 ==> "+(cnt_rejection[i]+1)+"회 거절");
    status = false;
}
}

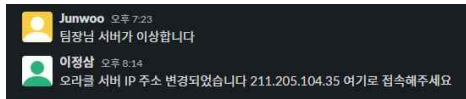
@@ -155,7 +155,7 @@ public class DietMakerSecond {
    }
    cur_cate cate_idx = fdDao
}
```

[illegible]

## 03 협업 - Slack

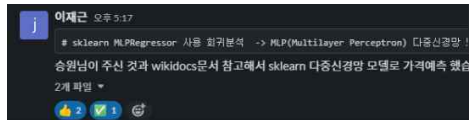
### 001 >> 업무 관련 소통 Slack 일원화

- 최신 정보 공유
- 프로젝트 진행상황 파악
- 업무 요청 및 응답 공통 채널 알림 활성화



### 002 >> 채널 기반 섹션 분할

- 각 섹션별 채널 운영
- 섹션별 코드 공유 및 코드 리뷰
- 섹션 담당자를 두어 섹션 위주의 의사소통 활성화



### 003 >> 데이터 및 정보 공유

- 데이터 및 참고 사이트 등의 링크를 공유
- 자료 위주의 섹션을 운영하여 검색 및 작업 효율성 제고



# 설계

01 구조 설계

02 DB 설계

03 화면 설계

# 01 구조 설계

## 데이터 수집

- 가격: API 활용 → [Price]
- 트렌드: 크롤링 → [Trend]
- 레시피: 크롤링 → [Recipe]
- 영양소: CSV data → [Food]
- 식단 : 논문 및 PDF 자료

> >

## 데이터 분석

- 가격: MLPRegressor
- 트렌드: Data 시각화(Bar)
- 레시피: 음식과 1대1 매칭
- 영양소: 데이터 1차처리
- 식단 : 추천 매커니즘 설계

> >

## 코드 구현

- 가격: 예측모델 생성
- 트렌드: 대시보드 chartjs
- 레시피: MVC 모델 생성
- 영양소: Food Dto 활용
- 식단 : 알고리즘 구현

> >

## 화면 구현

- 가격: 대시보드(Line)
- 트렌드: 대시보드(Bar)
- 레시피: 대시보드(Table)
- 영양소: 대시보드(Table)
- 식단 : 대시보드(App)

## 02 DB 설계

[illegible][illegible]

命令	说明	命令	说明	命令	说明
toad:acdb	创建数据库	toad:chgo	更改用户组	toad:cn	中文
toad:au	用户组	toad:err	错误	toad:en	英文
toad:db	数据库	toad:ex	导出	toad:ex	导出
toad:dbc	数据库连接	toad:ex2	导出2	toad:ex3	导出3
toad:dbf	数据库文件	toad:ex4	导出4	toad:ex5	导出5
toad:dbf2	数据库文件2	toad:ex6	导出6	toad:ex7	导出7
toad:dbf3	数据库文件3	toad:ex8	导出8	toad:ex9	导出9
toad:dbf4	数据库文件4	toad:ex10	导出10	toad:ex11	导出11
toad:dbf5	数据库文件5	toad:ex12	导出12	toad:ex13	导出13
toad:dbf6	数据库文件6	toad:ex14	导出14	toad:ex15	导出15
toad:dbf7	数据库文件7	toad:ex16	导出16	toad:ex17	导出17
toad:dbf8	数据库文件8	toad:ex18	导出18	toad:ex19	导出19
toad:dbf9	数据库文件9	toad:ex20	导出20	toad:ex21	导出21
toad:dbf10	数据库文件10	toad:ex22	导出22	toad:ex23	导出23
toad:dbf11	数据库文件11	toad:ex24	导出24	toad:ex25	导出25
toad:dbf12	数据库文件12	toad:ex26	导出26	toad:ex27	导出27
toad:dbf13	数据库文件13	toad:ex28	导出28	toad:ex29	导出29
toad:dbf14	数据库文件14	toad:ex30	导出30	toad:ex31	导出31
toad:dbf15	数据库文件15	toad:ex32	导出32	toad:ex33	导出33
toad:dbf16	数据库文件16	toad:ex34	导出34	toad:ex35	导出35
toad:dbf17	数据库文件17	toad:ex36	导出36	toad:ex37	导出37
toad:dbf18	数据库文件18	toad:ex38	导出38	toad:ex39	导出39
toad:dbf19	数据库文件19	toad:ex40	导出40	toad:ex41	导出41
toad:dbf20	数据库文件20	toad:ex42	导出42	toad:ex43	导出43
toad:dbf21	数据库文件21	toad:ex44	导出44	toad:ex45	导出45
toad:dbf22	数据库文件22	toad:ex46	导出46	toad:ex47	导出47
toad:dbf23	数据库文件23	toad:ex48	导出48	toad:ex49	导出49
toad:dbf24	数据库文件24	toad:ex50	导出50	toad:ex51	导出51
toad:dbf25	数据库文件25	toad:ex52	导出52	toad:ex53	导出53
toad:dbf26	数据库文件26	toad:ex54	导出54	toad:ex55	导出55
toad:dbf27	数据库文件27	toad:ex56	导出56	toad:ex57	导出57
toad:dbf28	数据库文件28	toad:ex58	导出58	toad:ex59	导出59
toad:dbf29	数据库文件29	toad:ex60	导出60	toad:ex61	导出61
toad:dbf30	数据库文件30	toad:ex62	导出62	toad:ex63	导出63
toad:dbf31	数据库文件31	toad:ex64	导出64	toad:ex65	导出65
toad:dbf32	数据库文件32	toad:ex66	导出66	toad:ex67	导出67
toad:dbf33	数据库文件33	toad:ex68	导出68	toad:ex69	导出69
toad:dbf34	数据库文件34	toad:ex70	导出70	toad:ex71	导出71
toad:dbf35	数据库文件35	toad:ex72	导出72	toad:ex73	导出73
toad:dbf36	数据库文件36	toad:ex74	导出74	toad:ex75	导出75
toad:dbf37	数据库文件37	toad:ex76	导出76	toad:ex77	导出77
toad:dbf38	数据库文件38	toad:ex78	导出78	toad:ex79	导出79
toad:dbf39	数据库文件39	toad:ex80	导出80	toad:ex81	导出81
toad:dbf40	数据库文件40	toad:ex82	导出82	toad:ex83	导出83
toad:dbf41	数据库文件41	toad:ex84	导出84	toad:ex85	导出85
toad:dbf42	数据库文件42	toad:ex86	导出86	toad:ex87	导出87
toad:dbf43	数据库文件43	toad:ex88	导出88	toad:ex89	导出89
toad:dbf44	数据库文件44	toad:ex90	导出90	toad:ex91	导出91
toad:dbf45	数据库文件45	toad:ex92	导出92	toad:ex93	导出93
toad:dbf46	数据库文件46	toad:ex94	导出94	toad:ex95	导出95
toad:dbf47	数据库文件47	toad:ex96	导出96	toad:ex97	导出97
toad:dbf48	数据库文件48	toad:ex98	导出98	toad:ex99	导出99
toad:dbf49	数据库文件49	toad:ex100	导出100	toad:ex101	导出101
toad:dbf50	数据库文件50	toad:ex102	导出102	toad:ex103	导出103
toad:dbf51	数据库文件51	toad:ex104	导出104	toad:ex105	导出105
toad:dbf52	数据库文件52	toad:ex106	导出106	toad:ex107	导出107
toad:dbf53	数据库文件53	toad:ex108	导出108	toad:ex109	导出109
toad:dbf54	数据库文件54	toad:ex110	导出110	toad:ex111	导出111
toad:dbf55	数据库文件55	toad:ex112	导出112	toad:ex113	导出113
toad:dbf56	数据库文件56	toad:ex114	导出114	toad:ex115	导出115
toad:dbf57	数据库文件57	toad:ex116	导出116	toad:ex117	导出117
toad:dbf58	数据库文件58	toad:ex118	导出118	toad:ex119	导出119
toad:dbf59	数据库文件59	toad:ex120	导出120		

[illegible][illegible]

변수명 (variable)	데이터 유형 (data type)	데이터 범위 (data range)	데이터 개수 (data count)
product_id	INTEGER	1~10	10
product_name	VARCHAR2	1~10	10
unit	VARCHAR2	1~10	10
price	NUMBER	1~10	10
price_percent	NUMBER	1~10	10
price_code	VARCHAR2	1~10	10

이름	타입	타입 설명	타입 범위
#policy_id	가계주번호	B/T	N
#date_issue	2025년 4월	DATE	N
#invoice_name	회사 이름	CHAR(50)	N
#id	아이디	CHAR(15)	N
#invoice_code	회사 코드	B/T	N
#invoice_code	회사 코드	CHAR(15)	N
#invoice	가계주 번호	FLD(4)	N

이름	타입	타입 설명	타입 범위
#id	회사주번호	B/T	N
#invoice_name	회사 이름	CHAR(50)	N
#invoice_code	회사 코드	CHAR(15)	N
#invoice_name	회사 이름	VARCHAR(50)	N
#invoice_inventories	회사주 번호	TEXT	N
#invoice_content	회사주 내용	TEXT	N
#invoice_code	회사 코드	CHAR(15)	N

이름	타입	타입 설명	타입 범위
#id	회사주번호	B/T	N
#invoice_name	회사 이름	CHAR(50)	N
#invoice_code	회사 코드	CHAR(15)	N
#invoice_name	회사 이름	VARCHAR(50)	N
#invoice_inventories	회사주 번호	TEXT	N
#invoice_content	회사주 내용	TEXT	N
#invoice_code	회사 코드	CHAR(15)	N

이름	타입	기본값	비고
send_id	varchar(10)		회원 ID
send_keyword	varchar(100)		검색 키워드
send_month	varchar(10)		월간 검색량
send_blog	varchar(100)		블로그 발행량
send_view	varchar(100)		커뮤니티 조회수
food_code	varchar(10)		음식 코드

[illegible]

notice			
공지 이름	공지 이름	varchar(255)	NOT NULL
notice_id	공지사항 번호	int	NOT NULL
notice_title	제목	varchar(255)	NOT NULL
notice_content	내용	text	NOT NULL
notice_writer	작성자	varchar(255)	NOT NULL
notice_date	작성일	date	NOT NULL

notice_type	공지사항 타입	notice_se	공지사항
notice_type	공지사항 타입	notice_se	공지사항

ALL  
ALL  
ALL  
ALL  
ALL  
ALL

- Table n

[illegible]

© HILL  
© HILL

## 주요 테이

name : food

## 주요 테이블 소개 - 음식

- Table name : food
- PK : food\_code
- 음식 및 재료 정보
- 34 cols x 11,862 rows



## 02 DB 설계

[illegible][illegible]

코드	이름	이름(영어)	이름(영어)의 로마자	이름(영어)의 한글
toosl_000	도스	도스	DOSE	도스
toosl_001	도스 000	도스 000	DOSE000	도스 000
toosl_002	도스 001	도스 001	DOSE001	도스 001
toosl_003	도스 002	도스 002	DOSE002	도스 002
toosl_004	도스 003	도스 003	DOSE003	도스 003
toosl_005	도스 004	도스 004	DOSE004	도스 004
toosl_006	도스 005	도스 005	DOSE005	도스 005
toosl_007	도스 006	도스 006	DOSE006	도스 006
toosl_008	도스 007	도스 007	DOSE007	도스 007
toosl_009	도스 008	도스 008	DOSE008	도스 008
toosl_010	도스 009	도스 009	DOSE009	도스 009
toosl_011	도스 010	도스 010	DOSE010	도스 010
toosl_012	도스 011	도스 011	DOSE011	도스 011
toosl_013	도스 012	도스 012	DOSE012	도스 012
toosl_014	도스 013	도스 013	DOSE013	도스 013
toosl_015	도스 014	도스 014	DOSE014	도스 014
toosl_016	도스 015	도스 015	DOSE015	도스 015
toosl_017	도스 016	도스 016	DOSE016	도스 016
toosl_018	도스 017	도스 017	DOSE017	도스 017
toosl_019	도스 018	도스 018	DOSE018	도스 018
toosl_020	도스 019	도스 019	DOSE019	도스 019
toosl_021	도스 020	도스 020	DOSE020	도스 020
toosl_022	도스 021	도스 021	DOSE021	도스 021
toosl_023	도스 022	도스 022	DOSE022	도스 022
toosl_024	도스 023	도스 023	DOSE023	도스 023
toosl_025	도스 024	도스 024	DOSE024	도스 024
toosl_026	도스 025	도스 025	DOSE025	도스 025
toosl_027	도스 026	도스 026	DOSE026	도스 026
toosl_028	도스 027	도스 027	DOSE027	도스 027
toosl_029	도스 028	도스 028	DOSE028	도스 028
toosl_030	도스 029	도스 029	DOSE029	도스 029
toosl_031	도스 030	도스 030	DOSE030	도스 030
toosl_032	도스 031	도스 031	DOSE031	도스 031
toosl_033	도스 032	도스 032	DOSE032	도스 032
toosl_034	도스 033	도스 033	DOSE033	도스 033
toosl_035	도스 034	도스 034	DOSE034	도스 034
toosl_036	도스 035	도스 035	DOSE035	도스 035
toosl_037	도스 036	도스 036	DOSE036	도스 036
toosl_038	도스 037	도스 037	DOSE037	도스 037
toosl_039	도스 038	도스 038	DOSE038	도스 038
toosl_040	도스 039	도스 039	DOSE039	도스 039
toosl_041	도스 040	도스 040	DOSE040	도스 040
toosl_042	도스 041	도스 041	DOSE041	도스 041
toosl_043	도스 042	도스 042	DOSE042	도스 042
toosl_044	도스 043	도스 043	DOSE043	도스 043
toosl_045	도스 044	도스 044	DOSE044	도스 044
toosl_046	도스 045	도스 045	DOSE045	도스 045
toosl_047	도스 046	도스 046	DOSE046	도스 046
toosl_048	도스 047	도스 047	DOSE047	도스 047
toosl_049	도스 048	도스 048	DOSE048	도스 048
toosl_050	도스 049	도스 049	DOSE049	도스 049
toosl_051	도스 050	도스 050	DOSE050	도스 050
toosl_052	도스 051	도스 051	DOSE051	도스 051
toosl_053	도스 052	도스 052	DOSE052	도스 052
toosl_054	도스 053	도스 053	DOSE053	도스 053
toosl_055	도스 054	도스 054	DOSE054	도스 054
toosl_056	도스 055	도스 055	DOSE055	도스 055
toosl_057	도스 056	도스 056	DOSE056	도스 056
toosl_058	도스 057	도스 057	DOSE057	도스 057
toosl_059	도스 058	도스 058	DOSE058	도스 058
toosl_060	도스 059	도스 059	DOSE059	도스 059
toosl_061	도스 060	도스 060	DOSE060	도스 060
toosl_062	도스 061	도스 061	DOSE061	도스 061
toosl_063	도스 062	도스 062	DOSE062	도스 062
toosl_064	도스 063	도스 063	DOSE063	도스 063
toosl_065	도스 064	도스 064	DOSE064	도스 064
toosl_066	도스 065	도스 065	DOSE065	도스 065
toosl_067	도스 066	도스 066	DOSE066	도스 066
toosl_068	도스 067	도스 067	DOSE067	도스 067
toosl_069	도스 068	도스 068	DOSE068	도스 068
toosl_070	도스 069	도스 069	DOSE069	도스 069
toosl_071	도스 070	도스 070	DOSE070	도스 070
toosl_072	도스 071	도스 071	DOSE071	도스 071
toosl_073	도스 072	도스 072	DOSE072	도스 072
toosl_074	도스 073	도스 073	DOSE073	도스 073
toosl_075	도스 074	도스 074	DOSE074	도스 074
toosl_076	도스 075	도스 075	DOSE075	도스 075
toosl_077	도스 076	도스 076	DOSE076	도스 076
toosl_078	도스 077	도스 077	DOSE077	도스 077
toosl_079	도스 078	도스 078	DOSE078	도스 078
toosl_080	도스 079	도스 079	DOSE079	도스 079
toosl_081	도스 080	도스 080	DOSE080	도스 080
toosl_082	도스 081	도스 081	DOSE081	도스 081
toosl_083	도스 082	도스 082	DOSE082	도스 082
toosl_084	도스 083	도스 083	DOSE083	도스 083
toosl_085	도스 084	도스 084	DOSE084	도스 084
toosl_086	도스 085	도스 085	DOSE085	도스 085
toosl_087	도스 086	도스 086	DOSE086	도스 086
toosl_088	도스 087	도스 087	DOSE087	도스 087
toosl_089	도스 088	도스 088	DOSE088	도스 088
toosl_090	도스 089	도스 089	DOSE089	도스 089
toosl_091	도스 090	도스 090	DOSE090	도스 090
toosl_092	도스 091	도스 091	DOSE091	도스 091
toosl_093	도스 092	도스 092	DOSE092	도스 092
toosl_094	도스 093	도스 093	DOSE093	도스 093
toosl_095	도스 094	도스 094	DOSE094	도스 094
toosl_096	도스 095	도스 095	DOSE095	도스 095
toosl_097	도스 096	도스 096	DOSE096	도스 096
toosl_098	도스 097	도스 097	DOSE097	도스 097
toosl_099	도스 098	도스 098	DOSE098	도스 098
toosl_100	도스 099	도스 099	DOSE099	도스 099
toosl_101	도스 100	도스 100	DOSE100	도스 100
toosl_102	도스 101	도스 101	DOSE101	도스 101
toosl_103	도스 102	도스 102	DOSE102	도스 102
toosl_104	도스 103	도스 103	DOSE103	도스 103
toosl_105	도스 104	도스 104	DOSE104	도스 104
toosl_106	도스 105	도스 105	DOSE105	도스 105
toosl_107	도스 106	도스 106	DOSE106	도스 106
toosl_108	도스 107	도스 107	DOSE107	도스 107
toosl_109	도스 108	도스 108	DOSE108	도스 108
toosl_110	도스 109	도스 109	DOSE109	도스 109
toosl_111	도스 110	도스 110	DOSE110	도스 110
toosl_112	도스 111	도스 111	DOSE111	도스 111
toosl_113	도스 112	도스 112	DOSE112	도스 112
toosl_114	도스 113	도스 113	DOSE113	도스 113
toosl_115	도스 114	도스 114	DOSE114	도스 114
toosl_116	도스 115	도스 115	DOSE115	도스 115
toosl_117	도스 116	도스 116	DOSE116	도스 116
toosl_118	도스 117	도스 117	DOSE117	도스 117
toosl_119	도스 118	도스 118	DOSE118	도스 118
toosl_120	도스 119	도스 119	DOSE119	도스 119
toosl_121	도스 120	도스 120	DOSE120	도스 120
toosl_122	도스 121	도스 121	DOSE121	도스 121
toosl_123	도스 122	도스 122	DOSE122	도스 122
toosl_124	도스 123	도스 123	DOSE123	도스 123
toosl_125	도스 124	도스 124	DOSE124	도스 124
toosl_126	도스 125	도스 125	DOSE125	도스 125
toosl_127	도스 126	도스 126	DOSE126	도스 126
toosl_128	도스 127	도스 127	DOSE127	도스 127
toosl_129	도스 128	도스 128	DOSE128	도스 128
toosl_130	도스 129	도스 129	DOSE129	도스 129
toosl_131	도스 130	도스 130	DOSE130	도스 130
toosl_132	도스 131	도스 131	DOSE131	도스 131
toosl_133	도스 132	도스 132	DOSE132	도스 132
toosl_134	도스 133	도스 133	DOSE133	도스 133
toosl_135	도스 134	도스 134	DOSE134	도스 134
toosl_136	도스 135	도스 135	DOSE135	도스 135
toosl_137	도스 136	도스 136	DOSE136	도스 136
toosl_138	도스 137	도스 137	DOSE137	도스 137
toosl_139	도스 138	도스 138	DOSE138	도스 138
toosl_140	도스 139	도스 139	DOSE139	도스 139
toosl_141	도스 140	도스 140	DOSE140	도스 140
toosl_142	도스 141	도스 141	DOSE141	도스 141
toosl_143	도스 142	도스 142	DOSE142	도스 142
toosl_144	도스 143	도스 143	DOSE143	도스 143
toosl_145	도스 144	도스 144	DOSE144	도스 144
toosl_146	도스 145	도스 145	DOSE145	도스 145
toosl_147	도스 146	도스 146	DOSE146	도스 146
toosl_148	도스 147	도스 147	DOSE147	도스 147
toosl_149	도스 148	도스 148	DOSE148	도스 148
toosl_150	도스 149	도스 149	DOSE149	도스 149
toosl_151	도스 150	도스 150	DOSE150	도스 150
toosl_152	도스 151	도스 151	DOSE151	도스 151
toosl_153	도스 152	도스 152	DOSE152	도스 152
toosl_154	도스 153	도스 153	DOSE153	도스 153
toosl_155	도스 154	도스 154	DOSE154	도스 154
toosl_156	도스 155	도스 155	DOSE155	도스 155
toosl_157	도스 156	도스 156	DOSE156	도스 156
toosl_158	도스 157	도스 157	DOSE157	도스 157
toosl_159	도스 158	도스 158	DOSE158	도스 158
toosl_160	도스 159	도스 159	DOSE159	도스 159
toosl_161	도스 160	도스 160	DOSE160	도스 160
toosl_162	도스 161	도스 161	DOSE161	도스 161
toosl_163	도스 162	도스 162	DOSE162	도스 162
toosl_164	도스 163	도스 163	DOSE163	도스 163
toosl_165	도스 164	도스 164	DOSE164	도스 164
toosl_166	도스 165	도스 165	DOSE165	도스 165
toosl_167	도스 166	도스 166	DOSE166	도스 166
toosl_168	도스 167	도스 167	DOSE167	도스 167
toosl_169	도스 168	도스 168	DOSE168	도스 168
toosl_170	도스 169	도스 169	DOSE169	도스 169
toosl_171	도스 170	도스 170	DOSE170	도스 170
toosl_172	도스 171	도스 171	DOSE171	도스 171
toosl_173	도스 172	도스 172	DOSE172	도스 172
toosl_174	도스 173	도스 173	DOSE173	도스 173
toosl_175	도스 174	도스 174	DOSE174	도스 174
toosl_176	도스 175	도스 175	DOSE175	도스 175
toosl_177	도스 176	도스 176	DOSE176	도스 176
toosl_178	도스 177	도스 177	DOSE177	도스 177
toosl_179	도스 178	도스 178	DOSE178	도스 178
toosl_180	도스 179	도스 179	DOSE179	도스 179
toosl_181	도스 180	도스 180	DOSE180	도스 180
toosl_182	도스 181	도스 181	DOSE181	도스 181
toosl_183	도스 182	도스 182	DOSE182	도스 182
toosl_184	도스 183	도스 183	DOSE183	도스 183
toosl_185	도스 184	도스 184	DOSE184	도스 184
toosl_186	도스 185	도스 185	DOSE185	도스 185
toosl_187	도스 186	도스 186	DOSE186	도스 186
toosl_188	도스 187	도스 187	DOSE187	도스 187
toosl_189	도스 188	도스 188	DOSE188	도스 188
toosl_190	도스 189	도스 189	DOSE189	도스 189
toosl_191	도스 190	도스 190	DOSE190	도스 190
toosl_192	도스 191	도스 191	DOSE191	도스 191
toosl_193	도스 192	도스 192	DOSE192	도스 192
toosl_194	도스 193	도스 193	DOSE193	도스 193
toosl_195	도스 194	도스 194	DOSE194	도스 194
toosl_196	도스 195	도스 195	DOSE195	도스 195
toosl_197	도스 196	도스 196	DOSE196	도스 196
toosl_198	도스 197	도스 197	DOSE197	도스 197
toosl_199	도스 198	도스 198	DOSE198	도스 198
toosl_200	도스 199	도스 199	DOSE199	도스 199
toosl_201	도스 200	도스 200	DOSE200	도스 200
toosl_202	도스 201	도스 201	DOSE201	도스 201
toosl_203	도스 202	도스 202	DOSE202	도스 202
toosl_204	도스 203	도스 203	DOSE203	도스 203
toosl_205	도스 204	도스 204	DOSE204	도스 204
toosl_206	도스 205	도스 205	DOSE205	도스 205
toosl_207	도스 206	도스 206	DOSE206	도스 206
toosl_208	도스 207	도스 207	DOSE207	도스 207
toosl_209	도스 208	도스 208	DOSE208	도스 208
toosl_210	도스 209	도스 209	DOSE209	도스 209
toosl_211	도스 210	도스 210	DOSE210	도스 210
toosl_212	도스 211	도스 211	DOSE211	도스

문자열의 처리 방법		문자열의 비교 방법	
strcmp	문자열 비교	strcmp	문자열 비교
strcpy	문자열 복사	strcmp	문자열 비교
strcat	문자열 연결	strcmp	문자열 비교
strncpy	문자열 복사 (길이 지정)	strcmp	문자열 비교
strncat	문자열 연결 (길이 지정)	strcmp	문자열 비교
strchr	문자열에서 문자 찾기	strcmp	문자열 비교
strrchr	문자열에서 문자 찾기 (뒤에서부터)	strcmp	문자열 비교
strstr	문자열에서 문자열 찾기	strcmp	문자열 비교
strtok	문자열 분할	strcmp	문자열 비교
strdup	문자열 복사 (동적 메모리)	strcmp	문자열 비교
strdupa	문자열 복사 (동적 메모리, 배열)	strcmp	문자열 비교
strdupf	문자열 복사 (동적 메모리, 파일)	strcmp	문자열 비교
strdupl	문자열 복사 (동적 메모리, 링크드 리스트)	strcmp	문자열 비교
strdupm	문자열 복사 (동적 메모리, 맵)	strcmp	문자열 비교
strdupn	문자열 복사 (동적 메모리, 배열, 링크드 리스트, 맵)	strcmp	문자열 비교
strdupo	문자열 복사 (동적 메모리, 오픈)	strcmp	문자열 비교
strdupp	문자열 복사 (동적 메모리, 프로세스)	strcmp	문자열 비교
strdupr	문자열 복사 (동적 메모리, 리눅스)	strcmp	문자열 비교
strdupu	문자열 복사 (동적 메모리, 유닉스)	strcmp	문자열 비교
strdupv	문자열 복사 (동적 메모리, 가상)	strcmp	문자열 비교
strdupw	문자열 복사 (동적 메모리, 윈도우)	strcmp	문자열 비교
strdupx	문자열 복사 (동적 메모리, X11)	strcmp	문자열 비교
strdupy	문자열 복사 (동적 메모리, Y16)	strcmp	문자열 비교
strdupz	문자열 복사 (동적 메모리, Z16)	strcmp	문자열 비교
strdupa	문자열 복사 (동적 메모리, 배열)	strcmp	문자열 비교
strdupf	문자열 복사 (동적 메모리, 파일)	strcmp	문자열 비교
strdupl	문자열 복사 (동적 메모리, 링크드 리스트)	strcmp	문자열 비교
strdupm	문자열 복사 (동적 메모리, 맵)	strcmp	문자열 비교
strdupn	문자열 복사 (동적 메모리, 배열, 링크드 리스트, 맵)	strcmp	문자열 비교
strdupo	문자열 복사 (동적 메모리, 오픈)	strcmp	문자열 비교
strdupp	문자열 복사 (동적 메모리, 프로세스)	strcmp	문자열 비교
strdupr	문자열 복사 (동적 메모리, 리눅스)	strcmp	문자열 비교
strdupu	문자열 복사 (동적 메모리, 유닉스)	strcmp	문자열 비교
strdupv	문자열 복사 (동적 메모리, 가상)	strcmp	문자열 비교
strdupw	문자열 복사 (동적 메모리, 윈도우)	strcmp	문자열 비교
strdupx	문자열 복사 (동적 메모리, X11)	strcmp	문자열 비교
strdupy	문자열 복사 (동적 메모리, Y16)	strcmp	문자열 비교
strdupz	문자열 복사 (동적 메모리, Z16)	strcmp	문자열 비교

변수명 (Variable)	타입 (Type)	값 (Value)
id	varchar(255)	1
username	varchar(255)	root
password	varchar(255)	123456
email	varchar(255)	root@localhost
phone	varchar(255)	1234567890
address	varchar(255)	1234567890
city	varchar(255)	1234567890
country	varchar(255)	1234567890
zip	varchar(255)	1234567890
status	varchar(255)	1
created_at	datetime	2023-01-01 12:00:00
updated_at	datetime	2023-01-01 12:00:00

변수명 (name)	변수 설명 (comment)	변수 타입 (type)	변수 값 (value)
pred_id	제품 ID	INT	111
pred_date	제품 출시 날짜	DATE	1611
food_name	음식 이름	CHAR(20)	1611
unit	단위	CHAR(10)	1611
pred_price	예측 가격	INT	1611
price_percent	가격 퍼센트	FLOAT	1611
food_cate	음식 카테고리	CHAR(10)	1611

price					dtype
price_kor	가격 정보	가격	INT		int64
price_date	가격 날짜	날짜	DATE		datetime64[ns]
food_name	음식 이름	이름	CHAR(50)		object
url	링크	URL	CHAR(150)		object
food_price	음식 가격	가격	INT		int64
food_taste	음식 맛	맛	CHAR(10)		object
price_percent	가격 변동률	변동률	FLOAT		float64

recipe_id	recipe_name	recipe_difficulty	recipe_cooking_time	recipe_serving_size
1	레시피 1	쉬움	30	2
2	레시피 2	중간	45	3
3	레시피 3	어려움	60	4
4	레시피 4	쉬움	20	1
5	레시피 5	중간	50	3
6	레시피 6	어려움	70	5
7	레시피 7	쉬움	15	1
8	레시피 8	중간	40	2
9	레시피 9	어려움	80	6
10	레시피 10	쉬움	25	1

트렌드	한국 이름	영어 이름	한국어 이름
#trend_id	트렌드 아이디	TREND_ID	
#trend_keyword	트렌드 키워드	TREND_KEYWORD	
#trend_month	트렌드 월	TREND_MONTH	
#trend_blog_id	트렌드 블로그 아이디	TREND_BLOG_ID	
#trend_view	트렌드 조회수	TREND_VIEW	
#trend_code	트렌드 코드	TREND_CODE	

변수명	변수 설명	변수 길이	변수 타입	변수 범위
id	회원 ID	10	VARCHAR(10)	NULL
name	회원 이름	20	VARCHAR(20)	NULL
email	회원 이메일	50	VARCHAR(50)	NULL
password	회원 비밀번호	20	VARCHAR(20)	NULL
phone	회원 전화번호	20	VARCHAR(20)	NULL
address	회원 주소	100	VARCHAR(100)	NULL
gender	회원 성별	1	CHAR(1)	NULL
birthday	회원 생일	10	VARCHAR(10)	NULL
profile	회원 프로필	1000	VARCHAR(1000)	NULL
status	회원 상태	1	CHAR(1)	NULL
created_at	회원 생성일	10	VARCHAR(10)	NULL
updated_at	회원更新日期	10	VARCHAR(10)	NULL

File	File 이름	Format	MD5 Hash	File Size
img	Image	RAW	INT	1024
img	Image	RAW	TEXT	1024
img	Image	RAW	TEXT	1024
img	Image	RAW	DATE	1024
img	Image	RAW	WAPCAP(1024)	1024
img	Image	RAW	WAPCAP(1024)	1024
img	Image	RAW	WAPCAP(1024)	1024

Notice	장
공지 이름	공지 이름
notice_id	공지사항 번호
notice_title	제목
notice_content	내용
notice_name	작성자
notice_date	작성일

notice_type	공지사항 타입	타입	VARCHAR(20)
notice_file	첨부파일	타입	VARCHAR(200)

Year	Total population aged 65 and older (millions)	Population aged 65 and older with disabilities (millions)
1980	18	4
1990	20	5
2000	23	7
2010	27	9
2020	38	12

- Table n°

## 주요 테이블 소개 - 영양소

**- Table name : nutrition**

- PK : nut\_id

- 연령 및 성별에 따른 필요 영양소(상한, 적정, 하한)

**- 85 cols x 9 rows**

## 02 DB 설계

項目	項目名稱	項目說明	項目類別	項目狀態
1	1.1	1.1.1	1.1.1.1	1.1.1.1
2	2.1	2.1.1	2.1.1.1	2.1.1.1
3	3.1	3.1.1	3.1.1.1	3.1.1.1
4	4.1	4.1.1	4.1.1.1	4.1.1.1
5	5.1	5.1.1	5.1.1.1	5.1.1.1
6	6.1	6.1.1	6.1.1.1	6.1.1.1
7	7.1	7.1.1	7.1.1.1	7.1.1.1
8	8.1	8.1.1	8.1.1.1	8.1.1.1
9	9.1	9.1.1	9.1.1.1	9.1.1.1
10	10.1	10.1.1	10.1.1.1	10.1.1.1
11	11.1	11.1.1	11.1.1.1	11.1.1.1
12	12.1	12.1.1	12.1.1.1	12.1.1.1
13	13.1	13.1.1	13.1.1.1	13.1.1.1
14	14.1	14.1.1	14.1.1.1	14.1.1.1
15	15.1	15.1.1	15.1.1.1	15.1.1.1
16	16.1	16.1.1	16.1.1.1	16.1.1.1
17	17.1	17.1.1	17.1.1.1	17.1.1.1
18	18.1	18.1.1	18.1.1.1	18.1.1.1
19	19.1	19.1.1	19.1.1.1	19.1.1.1
20	20.1	20.1.1	20.1.1.1	20.1.1.1
21	21.1	21.1.1	21.1.1.1	21.1.1.1
22	22.1	22.1.1	22.1.1.1	22.1.1.1
23	23.1	23.1.1	23.1.1.1	23.1.1.1
24	24.1	24.1.1	24.1.1.1	24.1.1.1
25	25.1	25.1.1	25.1.1.1	25.1.1.1
26	26.1	26.1.1	26.1.1.1	26.1.1.1
27	27.1	27.1.1	27.1.1.1	27.1.1.1
28	28.1	28.1.1	28.1.1.1	28.1.1.1
29	29.1	29.1.1	29.1.1.1	29.1.1.1
30	30.1	30.1.1	30.1.1.1	30.1.1.1
31	31.1	31.1.1	31.1.1.1	31.1.1.1
32	32.1	32.1.1	32.1.1.1	32.1.1.1
33	33.1	33.1.1	33.1.1.1	33.1.1.1
34	34.1	34.1.1	34.1.1.1	34.1.1.1
35	35.1	35.1.1	35.1.1.1	35.1.1.1
36	36.1	36.1.1	36.1.1.1	36.1.1.1
37	37.1	37.1.1	37.1.1.1	37.1.1.1
38	38.1	38.1.1	38.1.1.1	38.1.1.1
39	39.1	39.1.1	39.1.1.1	39.1.1.1
40	40.1	40.1.1	40.1.1.1	40.1.1.1
41	41.1	41.1.1	41.1.1.1	41.1.1.1
42	42.1	42.1.1	42.1.1.1	42.1.1.1
43	43.1	43.1.1	43.1.1.1	43.1.1.1
44	44.1	44.1.1	44.1.1.1	44.1.1.1
45	45.1	45.1.1	45.1.1.1	45.1.1.1
46	46.1	46.1.1	46.1.1.1	46.1.1.1
47	47.1	47.1.1	47.1.1.1	47.1.1.1
48	48.1	48.1.1	48.1.1.1	48.1.1.1
49	49.1	49.1.1	49.1.1.1	49.1.1.1
50	50.1	50.1.1	50.1.1.1	50.1.1.1
51	51.1	51.1.1	51.1.1.1	51.1.1.1
52	52.1	52.1.1	52.1.1.1	52.1.1.1
53	53.1	53.1.1	53.1.1.1	53.1.1.1
54	54.1	54.1.1	54.1.1.1	54.1.1.1
55	55.1	55.1.1	55.1.1.1	55.1.1.1
56	56.1	56.1.1	56.1.1.1	56.1.1.1
57	57.1	57.1.1	57.1.1.1	57.1.1.1
58	58.1	58.1.1	58.1.1.1	58.1.1.1
59	59.1	59.1.1	59.1.1.1	59.1.1.1
60	60.1	60.1.1	60.1.1.1	60.1.1.1
61	61.1	61.1.1	61.1.1.1	61.1.1.1
62	62.1	62.1.1	62.1.1.1	62.1.1.1
63	63.1	63.1.1	63.1.1.1	63.1.1.1
64	64.1	64.1.1	64.1.1.1	64.1.1.1
65	65.1	65.1.1	65.1.1.1	65.1.1.1
66	66.1	66.1.1	66.1.1.1	66.1.1.1
67	67.1	67.1.1	67.1.1.1	67.1.1.1
68	68.1	68.1.1	68.1.1.1	68.1.1.1
69	69.1	69.1.1	69.1.1.1	69.1.1.1
70	70.1	70.1.1	70.1.1.1	70.1.1.1
71	71.1	71.1.1	71.1.1.1	71.1.1.1
72	72.1	72.1.1	72.1.1.1	72.1.1.1
73	73.1	73.1.1	73.1.1.1	73.1.1.1
74	74.1	74.1.1	74.1.1.1	74.1.1.1
75	75.1	75.1.1	75.1.1.1	75.1.1.1
76	76.1	76.1.1	76.1.1.1	76.1.1.1
77	77.1	77.1.1	77.1.1.1	77.1.1.1
78	78.1	78.1.1	78.1.1.1	78.1.1.1
79	79.1	79.1.1	79.1.1.1	79.1.1.1
80	80.1	80.1.1	80.1.1.1	80.1.1.1
81	81.1	81.1.1	81.1.1.1	81.1.1.1
82	82.1	82.1.1	82.1.1.1	82.1.1.1
83	83.1	83.1.1	83.1.1.1	83.1.1.1
84	84.1	84.1.1	84.1.1.1	84.1.1.1
85	85.1	85.1.1	85.1.1.1	85.1.1.1
86	86.1	86.1.1	86.1.1.1	86.1.1.1
87	87.1	87.1.1	87.1.1.1	87.1.1.1
88	88.1	88.1.1	88.1.1.1	88.1.1.1
89	89.1	89.1.1	89.1.1.1	89.1.1.1
90	90.1	90.1.1	90.1.1.1	90.1.1.1
91	91.1	91.1.1	91.1.1.1	91.1.1.1
92	92.1	92.1.1	92.1.1.1	92.1.1.1
93	93.1	93.1.1	93.1.1.1	93.1.1.1
94	94.1	94.1.1	94.1.1.1	94.1.1.1
95	95.1	95.1.1	95.1.1.1	95.1.1.1
96	96.1	96.1.1	96.1.1.1	96.1.1.1
97	97.1	97.1.1	97.1.1.1	97.1.1.1
98	98.1	98.1.1	98.1.1.1	98.1.1.1
99	99.1	99.1.1	99.1.1.1	99.1.1.1
100	100.1	100.1.1	100.1.1.1	100.1.1.1

[illegible][illegible][illegible][illegible]

column name	data type	constraint
product_id	NUMBER(10,0)	PK
product_name	VARCHAR2(40)	
unit	VARCHAR2(10)	
price	NUMBER(10,2)	
discount	NUMBER(2,1)	
code	VARCHAR2(10)	

code	code	code	code	code	code
#price_low	가격 하한	price_low	가격 하한	INT	NUM
#price_high	가격 상한	price_high	가격 상한	DATE	NUM
#total_name	총합 이름	total_name	총합 이름	CHAR(30)	NUM
#name	이름	name	이름	CHAR(15)	NUM
#total_price	총합 가격	total_price	총합 가격	INT	NUM
#total_code	총합 코드	total_code	총합 코드	CHAR(12)	NUM
#price_percent	가격 백분율	price_percent	가격 백분율	FLOAT	NUM

code	code	code	code	code	code
#total_name	총합 이름	total_name	총합 이름	CHAR(30)	NUM
#price_code	가격 코드	price_code	가격 코드	CHAR(30)	NUM
#nick_name	닉네임	nick_name	닉네임	YAP(CHAR(15))	NUM
#nick_username	닉네임 사용자명	nick_username	닉네임 사용자명	TEXT	TEXT
#nick_content	닉네임 내용	nick_content	닉네임 내용	TEXT	TEXT
#total_code	총합 코드	total_code	총합 코드	CHAR(30)	NUM

code	code	code	code	code	code
#total_name	총합 이름	total_name	총합 이름	CHAR(30)	NUM
#price_code	가격 코드	price_code	가격 코드	CHAR(30)	NUM
#nick_name	닉네임	nick_name	닉네임	YAP(CHAR(15))	NUM
#nick_username	닉네임 사용자명	nick_username	닉네임 사용자명	TEXT	TEXT
#nick_content	닉네임 내용	nick_content	닉네임 내용	TEXT	TEXT
#total_code	총합 코드	total_code	총합 코드	CHAR(30)	NUM

변수 이름	종류 이름	해당된 데이터 값
→ <b>trwd_id</b>	<b>트레이드 번호</b>	60847
→ <b>trwd_keyword</b>	<b>트레이드 키워드</b>	TDCT
→ <b>trwd_month_val</b>	<b>월간 검색량</b>	60847
→ <b>trwd_tag_val</b>	<b>월호구 발행량</b>	60847
→ <b>trwd_view_val</b>	<b>월 발행량</b>	60847
→ <b>trwd_code</b>	<b>트레이드 코드</b>	CHP101

필수 속성	옵션 속성	데이터 타입	기본값
name	필수 속성	VARCHAR(128)	NULL
password	필수 속성	TEXT	NULL
email	선택 속성	VARCHAR(128)	NULL
phone	선택 속성	VARCHAR(128)	NULL
address	선택 속성	TEXT	NULL
name	선택 속성	VARCHAR(50)	NULL
password	선택 속성	TEXT	NULL
email	선택 속성	VARCHAR(128)	NULL

Field	Field Type	Field Size	Field Value
id	INT	4	1
question	TEXT	255	What is the capital of the United States?
answer	TEXT	255	Washington, D.C.
category	TEXT	255	General Knowledge
name	TEXT	255	John Doe
email	TEXT	255	john.doe@example.com
password	TEXT	255	12345678
created_at	DATETIME	8	2023-10-27 10:30:00
updated_at	DATETIME	8	2023-10-27 10:30:00

Source	Column	Table	Field	Value
모바일 앱	앱의 이름	앱명	INT	1
notice_id	공지사항 번호	notice_id	INT	1
notice_id	공지	notice_id	VARCHAR(128)	NULL
notice_content	내용	notice_content	TEXT	NULL
notice_name	제목	notice_name	VARCHAR(255)	NULL
notice_datetime	작성일	notice_datetime	DATE	NULL
notice_readnum	조회수	notice_readnum	INT	NULL
notice_type	공지사항 유형	notice_type	VARCHAR(255)	NULL
notice_file	첨부파일	notice_file	VARCHAR(255)	NULL

- Table n

## 주요 테이블 소개 - 가격

- Table name : price
- FK : food\_code
- 재료 가격에 대한 10년치 데이터
- 7 cols x 158,682 rows

## 02 DB 설계

[illegible][illegible]

命令	说明	命令	说明	命令	说明
toad:acdb	连接数据库	toad:chgo	更改语言	toad:nl	设置语言
toad:ansi	ANSI 模式	toad:exit	退出	toad:nl	设置语言
toad:ansi2ansi	ANSI 到 ANSI	toad:go	运行 SQL	toad:nl	设置语言
toad:ansi2ansi2	ANSI 到 ANSI2	toad:go2	运行 SQL2	toad:nl	设置语言
toad:ansi2ansi3	ANSI 到 ANSI3	toad:go3	运行 SQL3	toad:nl	设置语言
toad:ansi2ansi4	ANSI 到 ANSI4	toad:go4	运行 SQL4	toad:nl	设置语言
toad:ansi2ansi5	ANSI 到 ANSI5	toad:go5	运行 SQL5	toad:nl	设置语言
toad:ansi2ansi6	ANSI 到 ANSI6	toad:go6	运行 SQL6	toad:nl	设置语言
toad:ansi2ansi7	ANSI 到 ANSI7	toad:go7	运行 SQL7	toad:nl	设置语言
toad:ansi2ansi8	ANSI 到 ANSI8	toad:go8	运行 SQL8	toad:nl	设置语言
toad:ansi2ansi9	ANSI 到 ANSI9	toad:go9	运行 SQL9	toad:nl	设置语言
toad:ansi2ansi10	ANSI 到 ANSI10	toad:go10	运行 SQL10	toad:nl	设置语言
toad:ansi2ansi11	ANSI 到 ANSI11	toad:go11	运行 SQL11	toad:nl	设置语言
toad:ansi2ansi12	ANSI 到 ANSI12	toad:go12	运行 SQL12	toad:nl	设置语言
toad:ansi2ansi13	ANSI 到 ANSI13	toad:go13	运行 SQL13	toad:nl	设置语言
toad:ansi2ansi14	ANSI 到 ANSI14	toad:go14	运行 SQL14	toad:nl	设置语言
toad:ansi2ansi15	ANSI 到 ANSI15	toad:go15	运行 SQL15	toad:nl	设置语言
toad:ansi2ansi16	ANSI 到 ANSI16	toad:go16	运行 SQL16	toad:nl	设置语言
toad:ansi2ansi17	ANSI 到 ANSI17	toad:go17	运行 SQL17	toad:nl	设置语言
toad:ansi2ansi18	ANSI 到 ANSI18	toad:go18	运行 SQL18	toad:nl	设置语言
toad:ansi2ansi19	ANSI 到 ANSI19	toad:go19	运行 SQL19	toad:nl	设置语言
toad:ansi2ansi20	ANSI 到 ANSI20	toad:go20	运行 SQL20	toad:nl	设置语言
toad:ansi2ansi21	ANSI 到 ANSI21	toad:go21	运行 SQL21	toad:nl	设置语言
toad:ansi2ansi22	ANSI 到 ANSI22	toad:go22	运行 SQL22	toad:nl	设置语言
toad:ansi2ansi23	ANSI 到 ANSI23	toad:go23	运行 SQL23	toad:nl	设置语言
toad:ansi2ansi24	ANSI 到 ANSI24	toad:go24	运行 SQL24	toad:nl	设置语言
toad:ansi2ansi25	ANSI 到 ANSI25	toad:go25	运行 SQL25	toad:nl	设置语言
toad:ansi2ansi26	ANSI 到 ANSI26	toad:go26	运行 SQL26	toad:nl	设置语言
toad:ansi2ansi27	ANSI 到 ANSI27	toad:go27	运行 SQL27	toad:nl	设置语言
toad:ansi2ansi28	ANSI 到 ANSI28	toad:go28	运行 SQL28	toad:nl	设置语言
toad:ansi2ansi29	ANSI 到 ANSI29	toad:go29	运行 SQL29	toad:nl	设置语言
toad:ansi2ansi30	ANSI 到 ANSI30	toad:go30	运行 SQL30	toad:nl	设置语言
toad:ansi2ansi31	ANSI 到 ANSI31	toad:go31	运行 SQL31	toad:nl	设置语言
toad:ansi2ansi32	ANSI 到 ANSI32	toad:go32	运行 SQL32	toad:nl	设置语言
toad:ansi2ansi33	ANSI 到 ANSI33	toad:go33	运行 SQL33	toad:nl	设置语言
toad:ansi2ansi34	ANSI 到 ANSI34	toad:go34	运行 SQL34	toad:nl	设置语言
toad:ansi2ansi35	ANSI 到 ANSI35	toad:go35	运行 SQL35	toad:nl	设置语言
toad:ansi2ansi36	ANSI 到 ANSI36	toad:go36	运行 SQL36	toad:nl	设置语言
toad:ansi2ansi37	ANSI 到 ANSI37	toad:go37	运行 SQL37	toad:nl	设置语言
toad:ansi2ansi38	ANSI 到 ANSI38	toad:go38	运行 SQL38	toad:nl	设置语言
toad:ansi2ansi39	ANSI 到 ANSI39	toad:go39	运行 SQL39	toad:nl	设置语言
toad:ansi2ansi40	ANSI 到 ANSI40	toad:go40	运行 SQL40	toad:nl	设置语言
toad:ansi2ansi41	ANSI 到 ANSI41	toad:go41	运行 SQL41	toad:nl	设置语言
toad:ansi2ansi42	ANSI 到 ANSI42	toad:go42	运行 SQL42	toad:nl	设置语言
toad:ansi2ansi43	ANSI 到 ANSI43	toad:go43	运行 SQL43	toad:nl	设置语言
toad:ansi2ansi44	ANSI 到 ANSI44	toad:go44	运行 SQL44	toad:nl	设置语言
toad:ansi2ansi45	ANSI 到 ANSI45	toad:go45	运行 SQL45	toad:nl	设置语言
toad:ansi2ansi46	ANSI 到 ANSI46	toad:go46	运行 SQL46	toad:nl	设置语言
toad:ansi2ansi47	ANSI 到 ANSI47	toad:go47	运行 SQL47	toad:nl	设置语言
toad:ansi2ansi48	ANSI 到 ANSI48	toad:go48	运行 SQL48	toad:nl	设置语言
toad:ansi2ansi49	ANSI 到 ANSI49	toad:go49	运行 SQL49	toad:nl	设置语言
toad:ansi2ansi50	ANSI 到 ANSI50	toad:go50	运行 SQL50	toad:nl	设置语言
toad:ansi2ansi51	ANSI 到 ANSI51	toad:go51	运行 SQL51	toad:nl	设置语言
toad:ansi2ansi52	ANSI 到 ANSI52	toad:go52	运行 SQL52	toad:nl	设置语言
toad:ansi2ansi53	ANSI 到 ANSI53	toad:go53	运行 SQL53	toad:nl	设置语言
toad:ansi2ansi54	ANSI 到 ANSI54	toad:go54	运行 SQL54	toad:nl	设置语言
toad:ansi2ansi55	ANSI 到 ANSI55	toad:go55	运行 SQL55	toad:nl	设置语言
toad:ansi2ansi56	ANSI 到 ANSI56	toad:go56	运行 SQL56	toad:nl	设置语言
toad:ansi2ansi57	ANSI 到 ANSI57	toad:go57	运行 SQL57	toad:nl	设置语言
toad:ansi2ansi58	ANSI 到 ANSI58	toad:go58	运行 SQL58	toad:nl	设置语言
toad:ansi2ansi59	ANSI 到 ANSI59	toad:go59	运行 SQL59	toad:nl	设置语言
toad:ansi2ansi60	ANSI 到 ANSI60	toad:go60	运行 SQL60	toad:nl	设置语言

[illegible][illegible]

변수명 (variable)	변수 설명 (description)	변수 타입 (variable type)	변수 범위 (variable range)
pred_id	과거 판매번호	INT	1 ~ 1000
pred_date	과거 판매일자	DATE	2010-01-01 ~ 2010-12-31
food_name	음식 이름	CHAR(200)	1 ~ 200
unit	음식 단위	CHAR(10)	1 ~ 10
pred_price	과거 가격	INT	1 ~ 1000
price_percent	가격 변동률	FLOAT	1 ~ 100
food_code	음식 코드	CHAR(10)	1 ~ 10

기타 속성			
price_id	가격 정보	INT	NULL
price_date	가격 일자	DATE	NULL
food_name	음식 이름	CHAR(20)	NULL
unit	단위	CHAR(10)	NULL
food_price	음식 가격	INT	NULL
food_count	음식 개수	CHAR(10)	NULL

주문 정보			
id	주문 번호	INTEGER(10)	NOT NULL
order_name	주문자 이름	INT	
recipe_name	레시피 이름	CHAR(20)	
recipe_id	레시피 번호	CHAR(20)	
recipe_detail	레시피 내용	VARCHAR(100)	
recipeIngredients	레시피 재료	TEXT	
recipeContent	레시피 내용	TEXT	
recipePrice	주문 가격	CHAR(20)	

주문			
주문 번호	주문 일자	주문 내역	주문 가격

#food_code	음식 코드	CHWR101
------------	-------	---------

[illegible][illegible]

idname	idtype	idlength	idprecision	idscale
notice_id	문자열	10	0	0
notice_title	문자열	100	0	0
notice_content	문자열	1000	0	0
notice_name	문자열	100	0	0
notice_wednesday	문자열	10	0	0
notice_readnum	문자열	10	0	0
notice_type	문자열	100	0	0
notice_file	문자열	100	0	0

notice_type	공지사항 타입	111	VAR04ARE200
notice_file	공지사항 파일	111	VAR04ARE200

圖 14-1-10 鋼筋的彎曲

- Table n

## 주요 테이블 소개 - 레시피

- Table name : recipe
- FK : food\_code
- 음식에 대한 조리법(레시피) 정보
- 6 cols x 94,611 rows

## 02 DB 설계

項目	項目名稱	項目說明	項目類別	項目代碼
1	1.1	1.1.1	1.1.1.1	1.1.1.1.1
2	2.1	2.1.1	2.1.1.1	2.1.1.1.1
3	3.1	3.1.1	3.1.1.1	3.1.1.1.1
4	4.1	4.1.1	4.1.1.1	4.1.1.1.1
5	5.1	5.1.1	5.1.1.1	5.1.1.1.1
6	6.1	6.1.1	6.1.1.1	6.1.1.1.1
7	7.1	7.1.1	7.1.1.1	7.1.1.1.1
8	8.1	8.1.1	8.1.1.1	8.1.1.1.1
9	9.1	9.1.1	9.1.1.1	9.1.1.1.1
10	10.1	10.1.1	10.1.1.1	10.1.1.1.1
11	11.1	11.1.1	11.1.1.1	11.1.1.1.1
12	12.1	12.1.1	12.1.1.1	12.1.1.1.1
13	13.1	13.1.1	13.1.1.1	13.1.1.1.1
14	14.1	14.1.1	14.1.1.1	14.1.1.1.1
15	15.1	15.1.1	15.1.1.1	15.1.1.1.1
16	16.1	16.1.1	16.1.1.1	16.1.1.1.1
17	17.1	17.1.1	17.1.1.1	17.1.1.1.1
18	18.1	18.1.1	18.1.1.1	18.1.1.1.1
19	19.1	19.1.1	19.1.1.1	19.1.1.1.1
20	20.1	20.1.1	20.1.1.1	20.1.1.1.1
21	21.1	21.1.1	21.1.1.1	21.1.1.1.1
22	22.1	22.1.1	22.1.1.1	22.1.1.1.1
23	23.1	23.1.1	23.1.1.1	23.1.1.1.1
24	24.1	24.1.1	24.1.1.1	24.1.1.1.1
25	25.1	25.1.1	25.1.1.1	25.1.1.1.1
26	26.1	26.1.1	26.1.1.1	26.1.1.1.1
27	27.1	27.1.1	27.1.1.1	27.1.1.1.1
28	28.1	28.1.1	28.1.1.1	28.1.1.1.1
29	29.1	29.1.1	29.1.1.1	29.1.1.1.1
30	30.1	30.1.1	30.1.1.1	30.1.1.1.1
31	31.1	31.1.1	31.1.1.1	31.1.1.1.1
32	32.1	32.1.1	32.1.1.1	32.1.1.1.1
33	33.1	33.1.1	33.1.1.1	33.1.1.1.1
34	34.1	34.1.1	34.1.1.1	34.1.1.1.1
35	35.1	35.1.1	35.1.1.1	35.1.1.1.1
36	36.1	36.1.1	36.1.1.1	36.1.1.1.1
37	37.1	37.1.1	37.1.1.1	37.1.1.1.1
38	38.1	38.1.1	38.1.1.1	38.1.1.1.1
39	39.1	39.1.1	39.1.1.1	39.1.1.1.1
40	40.1	40.1.1	40.1.1.1	40.1.1.1.1
41	41.1	41.1.1	41.1.1.1	41.1.1.1.1
42	42.1	42.1.1	42.1.1.1	42.1.1.1.1
43	43.1	43.1.1	43.1.1.1	43.1.1.1.1
44	44.1	44.1.1	44.1.1.1	44.1.1.1.1
45	45.1	45.1.1	45.1.1.1	45.1.1.1.1
46	46.1	46.1.1	46.1.1.1	46.1.1.1.1
47	47.1	47.1.1	47.1.1.1	47.1.1.1.1
48	48.1	48.1.1	48.1.1.1	48.1.1.1.1
49	49.1	49.1.1	49.1.1.1	49.1.1.1.1
50	50.1	50.1.1	50.1.1.1	50.1.1.1.1
51	51.1	51.1.1	51.1.1.1	51.1.1.1.1
52	52.1	52.1.1	52.1.1.1	52.1.1.1.1
53	53.1	53.1.1	53.1.1.1	53.1.1.1.1
54	54.1	54.1.1	54.1.1.1	54.1.1.1.1
55	55.1	55.1.1	55.1.1.1	55.1.1.1.1
56	56.1	56.1.1	56.1.1.1	56.1.1.1.1
57	57.1	57.1.1	57.1.1.1	57.1.1.1.1
58	58.1	58.1.1	58.1.1.1	58.1.1.1.1
59	59.1	59.1.1	59.1.1.1	59.1.1.1.1
60	60.1	60.1.1	60.1.1.1	60.1.1.1.1
61	61.1	61.1.1	61.1.1.1	61.1.1.1.1
62	62.1	62.1.1	62.1.1.1	62.1.1.1.1
63	63.1	63.1.1	63.1.1.1	63.1.1.1.1
64	64.1	64.1.1	64.1.1.1	64.1.1.1.1
65	65.1	65.1.1	65.1.1.1	65.1.1.1.1
66	66.1	66.1.1	66.1.1.1	66.1.1.1.1
67	67.1	67.1.1	67.1.1.1	67.1.1.1.1

[illegible][illegible][illegible][illegible]

변수명 (variable)	변수 설명 (description)	데이터 타입 (data type)	데이터 범위 (data range)
pred_id	과자명 (candy name)	INT	1 ~ 10
pred_date	과자 구매 날짜 (candy purchase date)	DATE	2017-01-01 ~ 2017-12-31
food_name	음식명 (food name)	CHAR(20)	1 ~ 20
unit	음식 단위 (food unit)	CHAR(10)	1 ~ 10
pred_price	예측 가격 (predicted price)	INT	1 ~ 1000
price_percent	가격 변동률 (price change rate)	FLOAT	0.0 ~ 1.0
food_code	음식 코드 (food code)	CHAR(10)	1 ~ 10

속성	설명	타입	기본값
#price_low	가격 하한	INT	NULL
#price_high	가격 상한	DATE	NULL
#total_name	총합 이름	CHAR(30)	NULL
#total_price	총합 가격	INT	NULL
#total_code	총합 코드	CHAR(12)	NULL
#price_percent	가격 퍼센트	FLOAT	NULL

속성	설명	타입	기본값
#total_name	총합 이름	INT	NULL
#total_code	총합 코드	CHAR(30)	NULL
#total_price	총합 가격	CHAR(30)	NULL
#total_percent	총합 퍼센트	TEXT	NULL
#total_content	총합 내용	TEXT	NULL
#total_code	총합 코드	CHAR(30)	NULL

속성	설명	타입	기본값
#total_name	총합 이름	INT	NULL
#total_code	총합 코드	CHAR(30)	NULL
#total_price	총합 가격	CHAR(30)	NULL
#total_percent	총합 퍼센트	TEXT	NULL
#total_content	총합 내용	TEXT	NULL
#total_code	총합 코드	CHAR(30)	NULL

#	코드 이름	기능	비고
00	<code>word_id</code>	단어 ID	BIGIT
01	<code>word_jayword</code>	단어 감성도	TBIT
02	<code>word_month_voi</code>	월간 검색량	BIGIT
03	<code>word_day_voi</code>	일간 검색량	BIGIT
04	<code>word_view_voi</code>	뷰링량	BIGIT
05	<code>font_code</code>	글꼴 코드	CHEVET

필수 속성	옵션 속성	데이터 타입	기본값	비고
name	원래 테이블명	VARCHAR(128)	NULL	필수
username	사용자 이름	BIT	NULL	필수
email	이메일 주소	VARCHAR(128)	NULL	
phone	전화번호	VARCHAR(128)	NULL	
email	이메일	VARCHAR(128)	NULL	
name	이름	VARCHAR(128)	NULL	
password	비밀번호	DATE	NULL	
id	아이디	VARCHAR(128)	NULL	

notice			
== notice_id	공지사항 번호	varchar	8
+ notice_title	제목	varchar	100
+ notice_content	내용	text	
+ notice_name	작성자	varchar	20

• notocb_type	通知形態	100	WARDAREA
• notocb_se	部署形態	100	WARDAREA

Year	Actual (%)	Projected (%)
1980	11.5	-
1985	12.5	-
1990	13.5	-
1995	14.5	-
2000	15.0	-
2005	15.5	15.5
2010	15.5	16.5
2015	-	18.0
2020	-	19.5

- Table n

100

201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511	512	513	514	515	516	517	518	519	520	521	522	523	524	525	526	527	528	529	530	531	532	533	534	535	536	537	538	539	540	541	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559	560	561	562	563	564	565	566	567	568	569	570	571	572	573	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591	592	593	594	595	596	597	598	599	600	601	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	621	622	623	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639	640	641	642	643	644	645	646	647	648	649	650	651	652	653	654
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

**WILL  
O HELL  
SCHILL**

### 주요 테마

name : trend

## 주요 테이블 소개 - 트렌드

**- Table name : trend**

- FK : food\_code

### - 음식에 대한 월별 검색량 정보

**- 6 cols x 836 rows**

## 02 DB 설계

[illegible][illegible][illegible]

영문 이름	중국어 이름	조영준	김대호	정지민	정지현
week_11	주요 서신 발송	94.0	94.0	94.0	94.0
week_12	주한 의정 회담	94.0	94.0	94.0	94.0
week_10	주간 방송	94.0	94.0	94.0	94.0
foot_1	결혼	94.0	94.0	94.0	94.0
foot_2	죽음	94.0	94.0	94.0	94.0
foot_3	임신	94.0	94.0	94.0	94.0
foot_4	임산부	94.0	94.0	94.0	94.0
foot_5	임신	94.0	94.0	94.0	94.0
foot_6	부끄	94.0	94.0	94.0	94.0

[illegible]

이름	타입	길이	기본값	타입	길이	기본값
prod_id	제품 ID	NUMBER	10	PROD_ID	NUMBER	10
prod_name	제품명	VARCHAR2	40	PROD_NAME	VARCHAR2	40
unit	단위	VARCHAR2	10	UNIT	VARCHAR2	10
price	가격	NUMBER	8	PRICE	NUMBER	8
price_percent	가격 할인율	NUMBER	2	PRICE_PERCENT	NUMBER	2
prod_code	제품 코드	VARCHAR2	10	PROD_CODE	VARCHAR2	10

column	data type	length	precision	scale	nullable
price_id	NUMBER	4			NOT NULL
price_date	DATE				NULL
food_name	CHAR(30)				NULL
unit	CHAR(10)				NULL
food_price	NUMBER				NULL
#food_code	CHAR(10)				NULL
price_percent	NUMBER				NULL

이름	타입	길이	Nullable	Default Value
id	INTEGER	4	NOT NULL	0
food_name	CHAR	45	NOT NULL	
recipe_code	CHAR	10	NOT NULL	
recipe_name	CHAR	45	NOT NULL	
recipe_ingredients	TEXT		NOT NULL	
recipe_content	TEXT		NOT NULL	
#food_code	CHAR	10	NOT NULL	

Word			
종다 오름	전국 다름	내치서 사담	불
+ word_id	종근도 전오	POINT	NA
+ word_jayword	조표도 조표도	TEXT	NA
+ word_month_val	불근 조표도	POINT	NA
+ word_season_val	불조도 불조도	POINT	NA
+ word_view_val	귀 조표도	POINT	NA
# word_id	종근 조근	CUMEDIAN	

영어 이름	한글 이름	전화번호	이메일 주소	성별
Yasunori	야스노리 야다	03-3456-7890	YASUNORI@EXAMPLE.COM	M
Yumiko	유미코 야다	03-3456-7891	YUMIKO@EXAMPLE.COM	F
Yoshi	요시 야다	03-3456-7892	YOSHI@EXAMPLE.COM	M
Yuka	유카 야다	03-3456-7893	YUKA@EXAMPLE.COM	F
Yuzo	유조 야다	03-3456-7894	YUZO@EXAMPLE.COM	M
Yumi	유미 야다	03-3456-7895	YUMI@EXAMPLE.COM	F
Yoshiko	요시코 야다	03-3456-7896	YOSHIKO@EXAMPLE.COM	F
Yoshitaka	요시타카 야다	03-3456-7897	YOSHITAKA@EXAMPLE.COM	M
Yoshie	요시에 야다	03-3456-7898	YOSHIE@EXAMPLE.COM	F
Yoshinori	요시노리 야다	03-3456-7899	YOSHINORI@EXAMPLE.COM	M
Yoshitomo	요시토모 야다	03-3456-7900	YOSHITOMO@EXAMPLE.COM	M

[illegible]

Notice	공지사항	공지사항	공지사항	공지사항
공지사항	공지사항	공지사항	공지사항	공지사항
notice_id	공지사항 번호	INT	INT	INT
notice_title	제목	VARCHAR(100)	VARCHAR(100)	VARCHAR(100)
notice_content	내용	TEXT	TEXT	TEXT
notice_name	작성자	VARCHAR(20)	VARCHAR(20)	VARCHAR(20)
notice_wednesday	작성일	DATE	DATE	DATE
notice_readynum	조회수	INT	INT	INT
notice_status	공지사항 상태	VARCHAR(20)	VARCHAR(20)	VARCHAR(20)

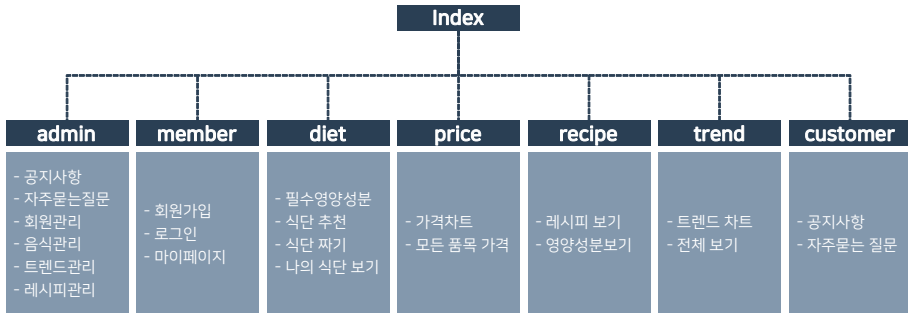
notice_type	공지사항 타입	NOTICE	VARCHAR(20) NULL
notice_file	첨부파일	FILE	VARCHAR(300) NULL

- member

## 그 외 테이블 소개

- member : 회원가입 통한 사용자 정보 저장
- notice : 관리자 의해 생성되는 공지사항 게시판
- custom\_diet : 사용자가 임의로 생성한 개인 식단
- week\_diet : 조건에 맞는 주간 추천 식단

## 03 화면 설계



## 구현

01 기본 환경

02 메인페이지

03 회원 및 관리자

04 가격

05 트렌드

06 영양소

07 레시피

08 식단

# 01 기본 환경

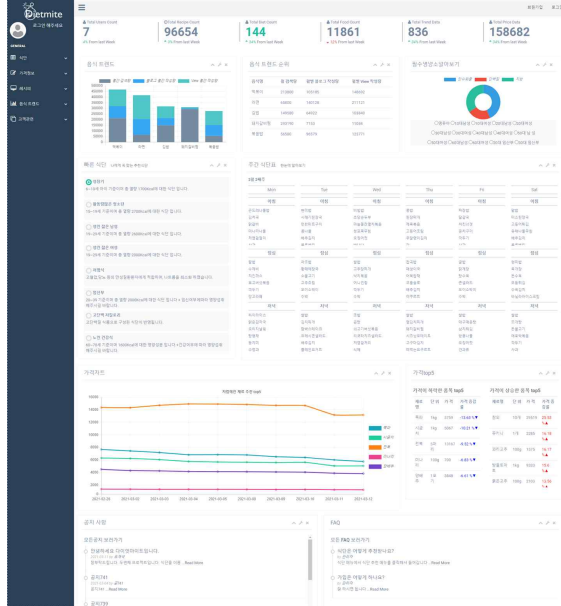
- **pom.xml** : dependancy → 라이브러리 추가
  - spring framework : spring MVC 프로젝트
  - mybatis : SQL → xml
  - mySqlConnection : DB 연결
  - cos : 파일 업로드
- web.xml : encoding, sitemesh 설정
- servlet-context : dataSource, sqlSession

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>kr.co</groupId>
  <artifactId>food</artifactId>
  <name>food</name>
  <packaging>war</packaging>
  <version>1.0.0-BUILD-SNAPSHOT</version>
  <properties>
    <java-version>1.6</java-version>
    <org.springframework-version>3.1.1.RELEASE</org.springframework-version>
    <org.aspectj-version>1.6.10</org.aspectj-version>
    <org.slf4j-version>1.6.6</org.slf4j-version>
  </properties>
  <dependencies>
    <!-- Spring -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-context</artifactId>
      <version>${org.springframework-version}</version>
      <exclusions>
        <!-- Exclude Commons Logging in favor of SLF4j -->
        <exclusion>
          <groupId>commons-logging</groupId>
          <artifactId>commons-logging</artifactId>
        </exclusion>
      </exclusions>
    </dependency>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-webmvc</artifactId>
      <version>${org.springframework-version}</version>
    </dependency>
```



# 02 메인페이지

- **index.jsp** : dash board
- **defalut.jsp** : <head> → meta, title, link
- **top\_menu** : bugger Button, 회원가입, 로그인
- **side\_menu** : profile, dropdown, navi
- **top tiles** : 전체사용자, 레시피, 식단, 음식, 가격
- **chart.js** → bar차트, pi차트, line 그래프



## 02 메인페이지

- index.jsp : dash board
- defalut.jsp : **<head>** → meta, title, link
- top\_menu : bugger Button, 회원가입, 로그인
- side\_menu : profile, dropdown, navi
- top tiles : 전체사용자, 레시피, 식단, 음식, 가격
- chart.js → bar차트, pi차트, line 그래프

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<!-- Meta, title, CSS, favicons, etc. -->
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">

<title>DietMite</title>

<!-- Bootstrap -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/vendor"
<!-- Font Awesome -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/vendor"
<!-- NProgress -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/vendor"
<!-- iCheck -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/vendor"
<!-- bootstrap-progressbar -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/vendor"
<!-- JQVMap -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/vendor"
<!-- bootstrap-daterangepicker -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/vendor"
<!-- Custom Theme Style -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/build/"
<!-- 파비콘 추가 -->
<link rel="shortcut icon" href="<%=request.getContextPath()%>/resour"
</head>
<decorator:head/>
<body class="nav-md">
  <div class="container body">
    <div class="main_container">
      <div class="col-md-3 left_col">
        <div class="left_col scroll-view">
          <div class="navbar nav_title" style="border: 0;">
            <a href="<%=request.getContextPath()%>/index.htm"
```

## 02 메인페이지

- index.jsp : dash board
- defalut.jsp : <head> → meta, title, link
- top\_menu : bugger Button, 회원가입, 로그인
- side\_menu : profile, dropdown, navi
- top tiles : 전체사용자, 레시피, 식단, 음식, 가격
- chart.js → bar차트, pi차트, line 그래프

```
<c:if test="${userid!='admin'}">
  <!-- sidebar menu -->
  <div id="sidebar-menu" class="main_menu_side hidden-print main_menu">
    <div class="menu_section">
      <h3>General</h3>
      <ul class="nav side-menu">
        <li><a><i class="fa fa-table"></i> 식단 <span
          class="fa fa-chevron-down"></span></a>
          <ul class="nav child_menu">
            <li><a
              href="<%=request.getContextPath()%>/nutri
            </li>
            <li><a href="/food/diet/reco_index">식단추천</a></li>
            <li><a href="/food/custom_diet/write">식단짜기</a></li>
            <li><a href="/food/custom_diet/list">나의 식단보기</li>
          </ul>
        </li>
        <li><a><i class="fa fa-edit"></i> 가격정보 <span
          class="fa fa-chevron-down"></span></a>
          <ul class="nav child_menu">
            <li><a href="/food/price/price_chart">가격 차트</a>
            <li><a href="/food/price/price_list">모든 품목 가격</li>
          </ul>
        </li>
        <li><a><i class="fa fa-desktop"></i> 레시피 <span
          class="fa fa-chevron-down"></span></a>
          <ul class="nav child_menu">
            <li><a href="/food/recipe_info/recipe_info_list">
            <li><a href="<%=request.getContextPath()%>/nutrit
              식품 영양성분보기</a></li>
          </ul>
        </li>
        <li><a><i class="fa fa-bar-chart-o"></i> 음식 트렌드 <span
          class="fa fa-chevron-down"></span></a>
          <ul class="nav child_menu">
```

## 02 메인페이지

- index.jsp : dash board
- defalut.jsp : <head> → meta, title, link
- top\_menu : bugger Button, 회원가입, 로그인
- side\_menu : profile, dropdown, navi
- top tiles : 전체사용자, 레시피, 식단, 음식, 가격
- **chart.js** → bar차트, pi차트, line 그래프

```
/* 나에게 필요한 필수영양소 비율 차트*/
Chart.defaults.global.maintainAspectRatio = false;
Chart.defaults.global.legend.labels.fontStyle = "italic";

var ctx = $("#pie-chart");
var pieLabels = ["탄수화물", "단백질", "지방"];
var pieData = [100, 50, 80];
var pieColors = [
    "#3498DB",
    "#E74C3C",
    "#26B99A",
];

var ctx2 = $("#pie-chart2");
var pieChart2 = new Chart(ctx2, {
    type: 'doughnut',
    data: {
        labels: pieLabels,
        datasets: [
            {
                data: pieData,
                backgroundColor: pieColors,
                borderColor: "transparent"
            }
        ]
    },
    options: {
        legend: {
            labels: {
                fontColor: "#73879C"
            }
        },
        cutoutPercentage: 50
    }
});
```

## 03 회원 및 관리자

- **회원가입** : 아이디 중복체크, 비밀번호 확인, 이메일 및 핸드폰 입력 체크
- 관리자 : 각종 게시판 CRUD 기능
- 로그인 : 비밀번호 체크
- 마이페이지 : 정보 수정 및 회원탈퇴 기능



회원가입 로그인

### 회원 가입

아이디

admin

사용 불가능한 아이디입니다

비밀번호

....

비밀번호 확인

....

이름

홍길동

이메일

hing@gmail.com

사용 가능한 이메일입니다

핸드폰

01020654032

나이

29

☒ 약관에 동의합니다

회원 가입



식단



가격정보



레시피



음식 트렌드



고객관련

## 03 회원 및 관리자

- 회원가입 : 아이디 중복체크, 비밀번호 확인, 이메일 및 핸드폰 입력 체크
- 관리자 : 각종 게시판 CRUD 기능
- 로그인 : 비밀번호 체크
- 마이페이지 : 정보 수정 및 회원탈퇴 기능

회원가입

로그인

로그인

아이디

admin

비밀번호

...

로그인

관리자

관리자

마이 페이지

아이디	admin
이름	관리자
이메일	admin@sarte.kr
핸드폰	01035077767
나이	34

정보 수정

회원 탈퇴

식단

가격정보

레시피

홈

공지사항

자주 묻는 질문

회원 관리

음식 관리

## 04 가격

- 데이터 수집 : KAMIS.org → API 요청 & json 리턴
- 데이터 저장 : csv File → MaridDB
- 예측모델 : scikit-learn → MLPRegressor
- 학습 데이터 : 10년치 재료별 일일 가격 데이터
- 활성화 함수 : Relu
- hidden layer size = (100, )
- score : 0.9866784
- 가격 차트 : chart.js의 line 그래프로 시각화

```
import requests
import pandas as pd
#DataFrame 생성
df=pd.DataFrame(columns=['date','item_name','kind_name','unit','price'])
# 날짜설정
# 일정 기간의 날짜를 얻어오는 함수
dt_index = pd.date_range(start='2021-03-05', end='2021-03-11',freq='B') #
# 일정 기간의 날짜를 str타입으로 치환하여 리스트에 담는다
dt_list = dt_index.strftime("%Y-%m-%d").tolist()
for date in dt_list:
    for j in [1,2,3,4,5,6]:
        item_code=j*100
        url="https://www.kamis.or.kr/service/price/xml.do?action=dailyPrice
            "&p_product_cls_code=01&convert_kg_yn=N" \
            "&p_cert_key=XXXX-XXXX-XXXX-XXXX&p_cert_id=XXXX&p_returntype=json
            "&p_regday="+date+"&p_item_category_code="+str(item_code)
        # api의 1번 "일별 부류별 도.소매가격정보" 사용함
        # url의 둘째줄 파라미터 -> 소매가, 전국기준(p_country_code 삭제), 단위변환
        # url의 셋째줄 파라미터 -> 개인키, 개인 아이디, 리턴타입=json
        # url의 넷째줄 파라미터 -> 날짜, 부류코드 순서
        # 여기에서 날짜와 부류코드를 변환해가면서 자료를 수집!
        # url을 불러와 json으로 data에 저장
        data=requests.get(url).json()
        # 그날의 시세 목록
        items=data['data']['item']
        for i in range(len(items)):
            # 상품명
            item_name=items[i]['item_name']
            # 상품 종류상세
            kind_name=items[i]['kind_name']
            # 상품단위
            unit=items[i]['unit']
            # 가격
            dpr1=items[i]['dpr1']
            price=dpr1.replace(',','')
            # df에 필요한 데이터(날짜, 상품명, 단위, 가격만 가져온다)
            df=df.append({'date':date, 'item_name':item_name, 'kind_name':kind_name, 'unit':unit, 'price':price})
            # 중복데이터 제거(상품명 중복된다면 첫번째 값만 남기기)
            df=df.drop_duplicates(['date','kind_name'], keep='first')
```

## 04 가격

- 데이터 수집 : KAMIS.org → API 요청 & json 리턴
- 데이터 저장 : csv File → MaridDB
- 예측모델 : scikit-learn → **MLPRegressor**
- 학습 데이터 : 10년치 재료별 일일 가격 데이터
- 활성화 함수 : Relu
- hidden layer size = (100, )
- score : 0.9866784
- 가격 차트 : chart.js의 line 그래프로 시각화

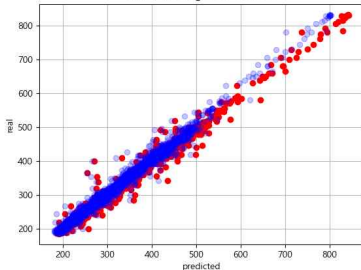
```
#sklearn MLPRegressor 사용 회귀분석 -> MLP(Multilayer Perceptron) 다중신
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from sklearn.neural_network import MLPRegressor
import pandas as pd
import numpy as np
# DB데이터 가져오기
data=pd.read_sql(sql,conn,parse_dates=['price_date'])
data=data.drop('price_id',axis=1)
data=data.drop('price_percent',axis=1)
# 예측값 테이블 만들기
pred_data=data[data['price_date']==nowday].reset_index() #nowday의 데이
pred_data=pred_data.drop('index',axis=1)
pred_data['price_date']=nextday
pred_data['food_price']='-'
# code_name의 값을 list로 뽑아오기
my_list=data['food_name'].values.tolist()
my_set=set(my_list)
my_list=list(my_set)
for li in my_list:
    data2=data[data['food_name']==li].reset_index()
    fin_data=np.array(data2['food_price'])
    xdata,ydata=makedata(fin_data)
    ydata=np.array(ydata).reshape(-1,1)
    #####데이터 전처리
    transformer = MinMaxScaler()
    transformer.fit(xdata)
    xdata_scaled = transformer.transform(xdata)
    trainx, testx, trainy, testy = train_test_split(xdata, ydata, test_
    #####모델 생성
    model = MLPRegressor(hidden_layer_sizes=(100, 100), activation='rel
    #####모델 학습
    model.fit(trainx, trainy)
    #####모델 검증
    # print(model.score(testx, testy))
    #####모델 예측
    # 예측하기 위해서는 x의 전날의 값 7개를 받아서 넣어야 된다.
    def date_7(date):
```



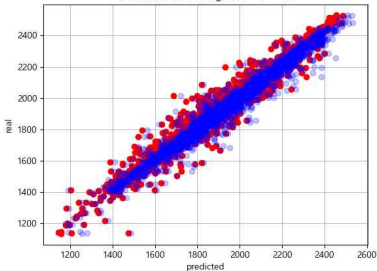
## 04 가격

- 데이터 수집 : KAMIS.org → API 요청 & json 리턴
- 데이터 저장 : csv File → MaridDB
- 예측모델 : scikit-learn → MLPRegressor
- 학습 데이터 : 10년치 재료별 일일 가격 데이터
- 활성화 함수 : Relu
- hidden layer size = (100, )
- score : 0.9866784
- 가격 차트 : chart.js의 line 그래프로 시각화

감자의 MLPRegressor 가격예측



돼지삼겹살의 MLPRegressor 가격예측



## 04 가격

- 데이터 수집 : KAMIS.org → API 요청 & json 리턴
- 데이터 저장 : csv File → MaridDB
- 예측모델 : scikit-learn → MLPRegressor
- 학습 데이터 : 10년치 재료별 일일 가격 데이터
- 활성화 함수 : Relu
- hidden layer size = (100, )
- score : 0.9866784
- 가격 차트 : chart.js의 line 그래프로 시각화



### 가격 차트



가격이 하락한 품목 top5

재료명	단위	가 격	가 격 증감률
목과	1kg	5759	-13.63 % ▼
시금지	1kg	5067	-10.21 % ▼
전복	5마리	13167	-9.52 % ▼
미나리	100g	700	-6.83 % ▼
양배추	1포기	3840	-6.61 % ▼

가격이 상승한 품목 top5

재료명	단위	가 격	가 격 증감률
참외	10개	29519	25.53 % ▲
쭈꾸미	1개	2285	16.18 % ▲
와리고추	100g	1575	16.17 % ▲
방울토마토	1kg	9333	15.6 % ▲
붉은고추	100g	2103	13.56 % ▲

## 05 트렌드

- 키워드 분석 사이트 → <https://blackkiwi.net/>
- 음식 키워드 크롤링 : PyAutoGui → 매크로 작성
- 자동완성 기능 : jQuery → autocomplete 함수
- 달력 구현 : java class 이용
- 트렌드 차트 : chart.js의 bar 그래프로 시각화

```
import pyautogui as pag
import time
import pandas as pd
import pyperclip
```

```
food = pd.read_csv('data/food.csv')
food = food[['DB군', '상용제품', '식품명']]
food = food[food['DB군']=='음식']
food = food[food['상용제품']=='품목대표']
food = food['식품명']
# print(food)
# print(food.shape)
food = food.reset_index(drop=True)
```

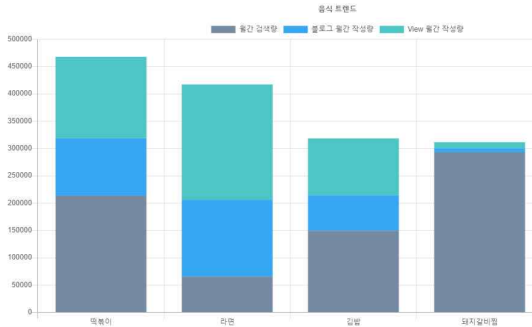
```
food2 = pd.read_csv('data/food2.csv')
print(food2)
food_remain = food[food.isin(food2['키워드'])==False]
print(food_remain)
food_remain = food_remain.reset_index(drop=True)
print(food_remain)
```

```
for i in range(len(food_remain)):
    pag.click(1700, 180)
    pyperclip.copy(food_remain[i])
    pag.hotkey("ctrl", "v")
    pag.press('enter')
    time.sleep(5)
    pag.doubleClick()
    pag.press('del')
```

## 05 트렌드

- 키워드 분석 사이트 → <https://blackkiwi.net/>
- 음식 키워드 크롤링 : PyAutoGui → 매크로 작성
- 자동완성 기능 : jQuery → autocomplete 함수
- 달력 구현 : java class 이용
- 트렌드 차트 : chart.js의 **bar 그래프**로 시각화

### 음식 트렌드



음식명	월간 검색량	월별 블로그 작성량	월별 View 작성량
떡볶이	213800	105185	148692
라면	65800	140128	211121
김밥	149580	64922	103840
돼지갈비찜	293190	7153	11084
뽕볶밥	56580	96579	123771
김밥	149580	64922	103840
돼지갈비찜	293190	7153	11084
뽕볶밥	56580	96579	123771

## 06 영양소

- 필수 영양소 DB 구축 및 list 구현
- Fetch API → 효율적 ajax 구현
- index 위치 조정 및 타 테이블과의 링크
- chart.js 활용 : 필수영양소 DB 시각화

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"
<!-- Meta, title, CSS, favicons, etc. -->
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/j
</script>
/*영양성분 content*/
function fetchPage(name){
    fetch(name).then(function(response){
        response.text().then(function(text){
            document.querySelector('nutrition').innerHTML = text
        })
    })
}
</script>
<!-- Bootstrap -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/
<!-- Font Awesome -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/
<!-- NProgress -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/
<!-- Custom Theme Style -->
<link href="<%=request.getContextPath()%>/resources/bootstrap/
</style>
table {
    width: 100%;
    border: 1px solid #444444;
    border-collapse: collapse;
}
th, td {
    border: 1px solid #444444;
    padding: 10px;
}
```

## 06 영양소

- 필수 영양소 DB 구축 및 list 구현
- Fetch API → 효율적 ajax 구현
- index 위치 조정 및 타 테이블과의 링크
- chart.js 활용 : 필수영양소 DB 시각화

Total Trend Data

836

▲ 34% From last Week

Total Price Data

158682

▲ 34% From last Week

필수영양소알아보기

원수화물 단백질 지방



○영유아 ○10대남성 ○10대여성 ○20대남성 ○20대여성

○30대남성 ○30대여성 ○40대남성 ○40대여성 ○50대 남성

○50대여성 ○60대남성 ○60대여성 ○20대 임신부 ○30대 임신부

Thu

아침

공밥  
된장찌개  
계곡볶음  
고등어조림  
무말랭이김치  
감

점심

잡곡밥  
매생이국  
어묵잔재

Fri

아침

짜장밥  
달걀국  
지킨너겟  
콩자구이  
각두기  
사과

점심

글밥  
닭계장  
참수육

Sat

아침

달걀  
미소된장국  
고등어튀김  
유채나물무침  
배추김치  
물루빅리

점심

현미밥  
육개장  
돈소육

# 07 레시피

- 레시피 참고 사이트 : 10000recipe.com
- 크롤링 → csv 파일 → DB 저장 (약 96,000 data)
- SQL join → 공통 컬럼 추출
- 식단에 포함된 음식명에 link → 레시피 연결
- 레시피 검색 기능 구현

```
# 아이들과 재료 크롤링 함수
def getTitleAndIngrediant(dom):
    cont_ingre = dom.find('div',class_='cont_ingre2')
    divs = cont_ingre.find_all('li')
    title = removeSpace(dom.find('div',class_="view2_summary st3").find('h3').text)
    i = 0
    ingre = []
    for div in divs:
        i += 1
        ingre.append(str(i)+''+removeSpace(div.text))
    return title, ingre
```

# 레시피 크롤링 함수

```
def getContent(dom):
    view_step = dom.find('div',class_='view_step')
    divs = view_step.find_all('div',{'class':"media-body"})
    i = 0
    contents = []
    for div in divs:
        i += 1
        contents.append(str(i)+''+ removeSpace(div.text))
    return contents
```

# 데이터프레임으로 데이터 생성하는 함수

```
def makeData(pageUrl, id_number):
    try:
        recvd = requests.get(pageUrl, verify=False, timeout=10)
        dom = BeautifulSoup(recvd.text,'lxml')
        title, r_ingre = getTitleAndIngrediant(dom)
        r_content = getContent(dom)

        df = pd.DataFrame({'id_number': [id_number], 'recipe_name' : [title], 'recipe_ingre': [r_ingre], 'recipe_content': [r_content]})
        if not os.path.exists(file_name):
            df.to_csv(file_name, index=False, mode='w', encoding='utf-8')
        else :
            df.to_csv(file_name, index=False, mode='a', encoding='utf-8', header = False)

    except AttributeError as e:
        pass
```

# 만개의 레시피 url 크롤링 함수

```
def main(url):
    recvd = requests.get(url, verify = False, timeout = 10)
    dom = BeautifulSoup(recvd.text,'lxml')
    lis = dom.find_all('li',class_='list-item')
    for li in lis:
```

# 07 레시피

- 레시피 참고 사이트 : 10000recipe.com
- 크롤링 → csv 파일 → DB 저장 (약 96,000 data)
- SQL join → 공통 컬럼 추출
- 식단에 포함된 음식명에 link → 레시피 연결
- 레시피 검색 기능 구현



## 레시피 전체 리스트

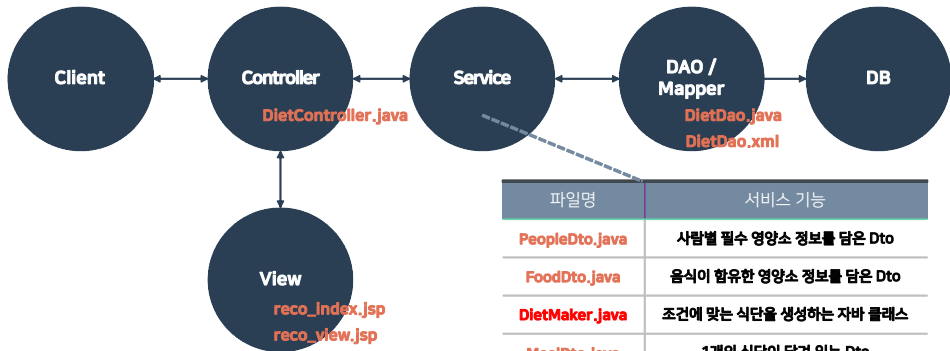
레시피 번호	레시피 이름
917867	매콤한 사전해물볶음밥
919938	비타민 가득한 피망볶음밥
919943	진밥으로 감자찰판구이밥
929678	매콤달콤 닭강정 삼각김밥
930399	뜨끈한 야채죽
932127	계란과 밥으로 만든 오프노미밥
950995	사골육수로 만든 간단백전 사골떡볶이
954422	선물로 좋은 달달한 단호박떡볶이
955234	러면보우 스킨롤
957180	선물용으로 좋은 삼색단자

이전 1 2 3 4 5 6 7 8 9 10 다음

레시피 재료	레시피 순서
1)사골육수2컵 2)조황이떡2컵 3)어묵1줄 4)물1큰술 5)통깨조금 6)고추장2큰술 7)사골육수2큰술 8)설탕1큰술 9)고춧가루1큰술 10)다진마늘1큰술 11)후추가루조금	1)먼저 양념장을 만들어 미리 숙성 시켜줍니다. 2)조황이떡(떡국 떡볶이떡이나 가래떡-먹기 좋은 크기로 썰어주세요~) 말랑한것은 그냥 쓰고 딱딱할 경우 미리 근한 물에 담구어 주세요~ 어묵은 뜨거운물을 부어 기름기를 제거 하고 먹기 좋은 크기로 썰어줍니다. 3)냄비나 찌꺼 사골육수를 넣고 향소금 끓여 미리 만들어 둔 양념장을 풀어주고향소금 끓여줍니다. 4)양념장이 끓으면 조황이떡과 어묵을 넣고 끓여주고 국물이 끓어 오르면 약한불로 줄여 걸죽해질 정도로 끓여 주면서 떡이 양념장과 고루 베이게 뒤섞어줍니다. 5)마지막에 꿀을 넣어 주고 물을 끈 후 찰시에 담아 통째 뿌려냅니다.



## 08 식단



파일명	서비스 기능
<code>PeopleDto.java</code>	사람별 필수 영양소 정보를 담은 Dto
<code>FoodDto.java</code>	음식이 함유한 영양소 정보를 담은 Dto
<code>DietMaker.java</code>	조건에 맞는 식단을 생성하는 자바 클래스
<code>MealDto.java</code>	1개의 식단이 담겨 있는 Dto
<code>People.java</code>	여러 식단을 보유한 사람 객체

## 08 식단

- 식단 추천 알고리즘 구현

- 1) 조건에 맞는 **영양성분표** 가져오기
- 2) false인 음식 꾸러미 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → 거절횟수 업데이트
- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

	칼로리	단백질	...	칼슘	아연
사람A <b>하한선</b>	300	400	...	30	20
사람A <b>적정선</b>	500	500	...	50	50
사람A <b>상한선</b>	700	600	...	70	80

## 08 식단

- 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기
- 2) false인 **음식 꾸러미** 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → 거절횟수 업데이트
- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 false	반찬2 false	반찬3 false
주식 false	부식 false	국물 false

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	0	0	...	0	0



## 08 식단

- 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기

- 2) false인 음식 꾸러미 가져오기

- 3) 음식 하나 꺼내기

- a. 상한선 이하 → 음식 추가 → 2)

- b. 상한선 초과 → 거절횟수 업데이트

- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)

- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 false	반찬2 false	반찬3 false
주식 false	부식 false	국물 false

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	0	0	...	0	0

소갈비찜

200	300	...	20	10
-----	-----	-----	----	----



## 08 식단

- 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기

- 2) false인 음식 꾸러미 가져오기

- 3) 음식 하나 꺼내기

- a. 상한선 이하 → 음식 추가 → 2)

- b. 상한선 초과 → 거절횟수 업데이트

- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)

- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	반찬2 false	반찬3 false	소갈비찜		
주식 false	부식 false	국물 false			

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	200	300	...	20	10

## 08 식단

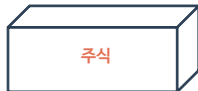
### • 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기
- 2) false인 **음식 꾸러미** 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → 거절횟수 업데이트
- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	반찬2 false	반찬3 false	소갈비찜		
<b>주식</b> false	부식 false	국물 false			

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	200	300	...	20	10



## 08 식단

### • 식단 추천 알고리즘 구현

1) 조건에 맞는 영양성분표 가져오기

2) false인 음식 꾸러미 가져오기

3) 음식 하나 꺼내기

a. 상한선 이하 → 음식 추가 → 2)

b. 상한선 초과 → 거절횟수 업데이트

4) if 4회 거절 → 최고 수치 음식 제거 → 2)

5) 음식 가짓수 만족시까지 2) ~ 4) 반복

• 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	반찬2 false	반찬3 false	소갈비찜		
주식 false	부식 false	국물 false			

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	200	300	...	20	10

짜장밥

600	200	...	30	20
-----	-----	-----	----	----



## 08 식단

### • 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기
- 2) false인 음식 꾸러미 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → 거절횟수 업데이트
- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	반찬2 false	반찬3 false	소갈비찜		
주식 false	부식 false	국물 false			

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	200	300	...	20	10

짜장밥	600	200	...	30	20
-----	-----	-----	-----	----	----





## 08 식단

### • 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기
- 2) false인 음식 꾸러미 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → **거절횟수 업데이트**
- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

### • 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	반찬2 false	반찬3 false	소갈비찜		
주식 false	부식 false	국물 false			

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	200	300	...	20	10
<b>거절횟수</b>	<b>1</b>	0	...	0	0

## 08 식단

### • 식단 추천 알고리즘 구현

1) 조건에 맞는 영양성분표 가져오기

2) false인 **음식 꾸러미** 가져오기

3) **음식 하나 꺼내기**

a. 상한선 이하 → 음식 추가 → 2)

b. 상한선 초과 → 거절횟수 업데이트

4) if 4회 거절 → 최고 수치 음식 제거 → 2)

5) 음식 가짓수 만족시까지 2) ~ 4) 반복

• 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	반찬2 false	반찬3 false	소갈비찜		
<b>주식</b> false	부식 false	국물 false			

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	200	300	...	20	10
거절횟수	1	0	...	0	0
<b>잡곡밥</b>	300	100	...	20	30



## 08 식단

### • 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기
- 2) false인 음식 꾸러미 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → 거절횟수 업데이트
- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	반찬2 false	반찬3 false	소갈비찜		
주식 true	부식 false	국물 false	잡곡밥		

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	500	400	...	30	40
거절횟수	1	0	...	0	0
잡곡밥	300	100	...	20	30



## 08 식단

### • 식단 추천 알고리즘 구현

1) 조건에 맞는 영양성분표 가져오기

2) false인 **음식 꾸러미** 가져오기

3) **음식 하나 꺼내기**

a. 상한선 이하 → 음식 추가 → 2)

b. **상한선 초과** → **거절횟수 업데이트**

4) if 4회 거절 → 최고 수치 음식 제거 → 2)

5) 음식 가짓수 만족시까지 2) ~ 4) 반복

• 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	<b>반찬2</b> false	반찬3 false	소갈비찜		
주식 true	부식 false	국물 false	잡곡밥		

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	<b>700</b>	600	...	70	<b>80</b>
현재영양소	<b>500</b>	400	...	30	<b>40</b>
거절횟수	<b>2</b>	0	...	0	<b>1</b>

<b>잡채</b>	250	100	...	10	<b>50</b>
-----------	-----	-----	-----	----	-----------



## 08 식단

### • 식단 추천 알고리즘 구현

1) 조건에 맞는 영양성분표 가져오기

2) false인 음식 꾸러미 가져오기

3) 음식 하나 꺼내기

a. 상한선 이하 → 음식 추가 → 2)

b. 상한선 초과 → 거절횟수 업데이트

4) if 4회 거절 → 최고 수치 음식 제거 → 2)

5) 음식 가짓수 만족시까지 2) ~ 4) 반복

• 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	반찬2 false	반찬3 false	소갈비찜		
주식 true	부식 false	국물 false	잡곡밥		

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	500	400	...	30	40
거절횟수	3	1	...	0	1
소세지볶음	450	300	...	10	10



## 08 식단

### • 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기
- 2) false인 음식 꾸러미 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → 거절횟수 업데이트
- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	반찬2 false	반찬3 false	소갈비찜		
주식 true	부식 false	국물 false	잡곡밥		

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	500	400	...	30	40
거절횟수	4	1	...	0	1
어묵조림	300	100	...	20	20



## 08 식단

- 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기
- 2) false인 음식 꾸러미 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → 거절횟수 업데이트
- 4) if 4회 거절 → 최고 수치 **음식 제거** → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 false	반찬2 false	반찬3 false			
주식 true	부식 false	국물 false	잡곡밥		

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	300	100	...	10	30
거절횟수	0	1	...	0	1

## 08 식단

### • 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기
- 2) false인 음식 꾸러미 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → 거절횟수 업데이트
- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 false	반찬2 false	반찬3 false			
주식 true	부식 false	국물 false	잡곡밥		

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	300	100	...	10	30
거절횟수	0	1	...	0	1
고등어구이	100	300	...	20	10





## 08 식단

- 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기
- 2) false인 음식 꾸러미 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → 거절횟수 업데이트
- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

반찬1 true	반찬2 true	반찬3 true	고등어 구이	감자조림	배추김치
주식 true	부식 true	국물 true	잡곡밥	요구르트	근대 된장국

	칼로리	단백질	...	칼슘	아연
사람A 하한선	300	400	...	30	20
사람A 적정선	500	500	...	50	50
사람A 상한선	700	600	...	70	80
현재영양소	650	500	...	60	50

## 08 식단

### • 식단 추천 알고리즘 구현

- 1) 조건에 맞는 영양성분표 가져오기
- 2) false인 음식 꾸러미 가져오기
- 3) 음식 하나 꺼내기
  - a. 상한선 이하 → 음식 추가 → 2)
  - b. 상한선 초과 → 거절횟수 업데이트
- 4) if 4회 거절 → 최고 수치 음식 제거 → 2)
- 5) 음식 가짓수 만족시까지 2) ~ 4) 반복

### • 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

```
public static ArrayList<FoodDto> get_meal(People people, DietDao ddao) {
    food_list = get_100_foods(ddao);
    int i = 0;
    while(true) {
        if (i==10000) break;
        if(is_finish()) {
            out_meal();
            break;
        }
        check_nut(peoples.nut_lb, cur_nut, peoples.nut_ub, ddao);
        i += 1;
        System.out.println(i+"번 try++++++++++++++++++++++++++++++++");
    }
    return meals;
}

public static boolean check_nut(double[] nut_lb, double[] cur_nut, double[]
for (int i=0; i<nut_lb.length; i++) {
    if(cur_nut[i]>= nut_lb[i]) {
        continue;
    }
    FoodDto food = get_food(i, ddao);
    if(is_under_ub(nut_ub, food)) {
        System.out.println("영양성분 만족 ===== ["+food.getFood_cate3()+"]");
        add_food(food, ddao);
        return true;
    } else if(meals.size()==0) {
        del_from_foodlist(food, cate_idx, ddao);
    }
}
return false;
}

public static void add_rejection(int nut_idx) {
    cnt_rejection[nut_idx] += 1;
    if(cnt_rejection[nut_idx]==4) {
        if(meals.size()==0) {
            cnt_rejection = new int[27];
            return;
        }
        double max_value = 0.0;
        int max_idx = 0;
        for (int i=0; i< meals.size(); i++) {
            if (getFoodNutByIndex(meals.get(i), nut_idx) > max_value) {
                max_value = getFoodNutByIndex(meals.get(i), nut_idx);
                max_idx = i;
            }
        }
        del_from_foodlist(meals.get(max_idx), cate_idx, ddao);
    }
}
```

## 08 식단

- 식단 추천 알고리즘 구현

- 조건에 맞는 영양성분표 가져오기
- false인 음식 꾸러미 가져오기
- 음식 하나 꺼내기
  - 상한선 이하 → 음식 추가 → 2)
  - 상한선 초과 → 거절횟수 업데이트
- if 4회 거절 → 최고 수치 음식 제거 → 2)
- 음식 가짓수 만족시까지 2) ~ 4) 반복

- 추후 조건 추가 → DTO만 업데이트 → 코드 변동 X

```
Console x
Pivotal tc Server Developer Edition v4.0 (Pivotal tc Server) C:\Program Files\Java\jdk-14.0.2\bin\javaw.exe (2021)
마그네슘, 4회 초과=> <업무열거미김치> 삭제
마그네슘 초과 ==> 0회 거절
아연 초과 ==> 1회 거절
영양성분 만족 ==> [주식] 비빔밥
23번 try+++++++
칼로리 초과 ==> 1회 거절
아연 초과 ==> 2회 거절
엽산 초과 ==> 2회 거절
마그네슘 초과 ==> 1회 거절
아연 초과 ==> 3회 거절
구리 초과 ==> 1회 거절
마그네슘 초과 ==> 2회 거절
아연, 4회 초과=> <비빔밥> 삭제
아연 초과 ==> 0회 거절
영양성분 만족 ==> [주식] 송이덮밥
24번 try+++++++
칼로리 초과 ==> 2회 거절
마그네슘 초과 ==> 3회 거절
아연 초과 ==> 1회 거절
구리 초과 ==> 2회 거절
영양성분 만족 ==> [국물] 돼지고기김치찌개
25번 try+++++++
영양성분 만족 ==> [반찬3] 무말랭이 김치
26번 try+++++++
칼로리 초과 ==> 3회 거절
칼로리, 4회 초과=> <송이덮밥> 삭제
칼로리 초과 ==> 0회 거절
영양성분 만족 ==> [주식] 기장밥
27번 try+++++++
영양성분 만족 ==> [부식] 플레인(컵)
28번 try+++++++
=====오늘의 식단=====
[반찬1] 너겟킹(4조각)
[반찬2] 양파볶음
[국물] 돼지고기김치찌개
[반찬3] 무말랭이 김치
[주식] 기장밥
[부식] 플레인(컵)
```

# 시연

01 현장 시연

02 QnA

03 레퍼런스

# 시연

01 현장 시연

02 QnA

03 레퍼런스

## 03 레퍼런스

식약처에서 제공하는 건강기능식품으로 인정받은 보충제나 영양제

[https://foodsafetykorea.go.kr/api/datasetList.do?svc\\_type\\_cd=API\\_TYPE06  
&menu\\_grp=MENU\\_GRP31&menu\\_no=661](https://foodsafetykorea.go.kr/api/datasetList.do?svc_type_cd=API_TYPE06&menu_grp=MENU_GRP31&menu_no=661)

빅데이터 분석을 통한 농업 정보

<http://www.nongsaro.go.kr/portal/ps/psv/psvr/psvre/curationDtl.ps?menuId=PS03352&srchCurationNo=1123&pageUnit=6&srchStr=>

농산물유통정보

<https://www.kamis.or.kr/customer/price/wholesale/item.do>

영양학에 관련된 학술지

[https://scholarworks.gvsu.edu/cgi/viewcontent.cgi?article=1068&context=oapfs\\_articles](https://scholarworks.gvsu.edu/cgi/viewcontent.cgi?article=1068&context=oapfs_articles)

농식품종합정보시스템 - 식단관리(메뉴젠) 식단작성

<http://koreanfood.rda.go.kr/kfi/openapi/useNewGuidance>

농산물유통정보

<https://www.kamis.or.kr/customer/price/wholesale/period.do>

농림식품산업 미래성장포럼

<https://www.bioin.or.kr/InnoDS/data/upload/industry/aaf70a0218fb47c38522860aa9cbc11c.pdf>

축산물품질평가원 축산유통정보

<https://www.ekapepia.com/index.do>

스마트하게 식단을 관리하는 딥러닝 기술 정보

<https://tech.kakaoenterprise.com/84>

식이요법 전문 연구기업 닥터키친

<https://doctorkitchen.co.kr/programs/?s=treatment&w=2&d=0>

공공기관 레시피 API

<https://www.foodsafetykorea.go.kr/api/newDatasetDetail.do>

공공기관 식품안전나라

<https://www.foodsafetykorea.go.kr/main.do>

색 구성표 사이트

<https://colorate.azurewebsites.net/ko/Color/2A3F54>

MLPregressor 학술지

[http://www.gabormelli.com/RKB/sklearn.neural\\_network.MLPRegressor](http://www.gabormelli.com/RKB/sklearn.neural_network.MLPRegressor)

Chart.js 참고 사이트

<https://www.chartjs.org/samples/latest>

식단 추천 알고리즘 관련 논문

[https://scholarworks.gvsu.edu/cgi/viewcontent.cgi?article=1068&context=oapfs\\_articles](https://scholarworks.gvsu.edu/cgi/viewcontent.cgi?article=1068&context=oapfs_articles)

**감사합니다**