

멋쟁이 사자처럼 at 이화여대

Web / Ruby / Rails

3 / 27 (화)



LIKE LION

Part 1 . Ruby



LIKE LION



High-Level

사람 말에 가깝고

Interpreted

이해하기 쉽고

Object-oriented

상식적이고

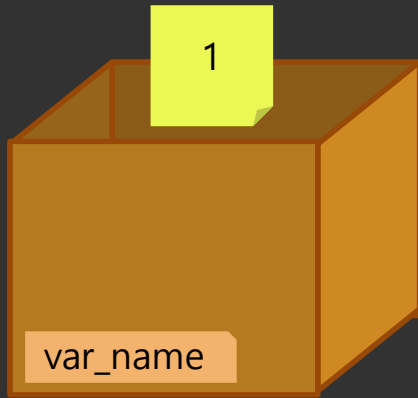
Easy to use

쉽다



LIKE LION

변수 (Variable)



number
string
true / false

정보를 담는 박스

var name = content

content라는 내용이 var_name이라는 이름으로 저장되었다.

변수 선언 : 변수의 이름만 적으면 된다.

> var_name

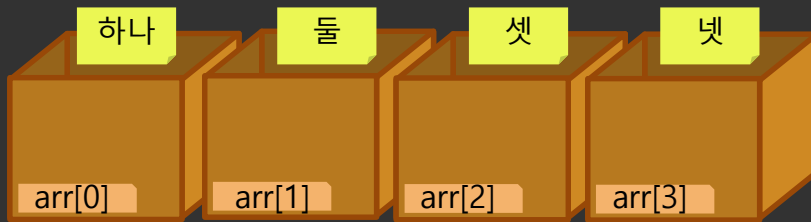
변수 할당 : =을 사용해서 변수에 값(내용)을 부여하는 것

> var_name = 1



LIKE LION

배열 (Array)



박스들의 묶음

`arr = [con1, con2, con3, ...]`

같은 자료형을 가지는 것들 끼리 묶는다. (숫자, 문자열 ...)

배열안에 있는 각 요소에는 **index**가 부여된다.

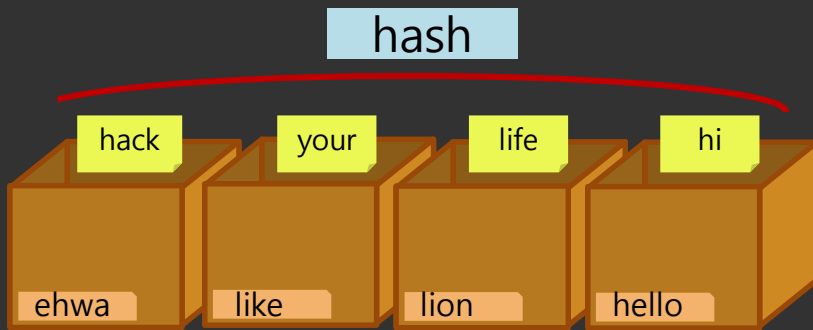
> `arr = ["하나", "둘", "셋", ...]`

> `arr[0] = "하나" arr[1] = "둘" arr[2] = "셋"`



LIKE LION

해시 (Hash)



이름이 지정된 박스들의 묶음

```
hash = {  
    key1 => value1,  
    key2 => value2  
}
```

```
> hash = {"ewha" => "hack", "like" => "your",  
          "lion" => "life", "hello" => "hi"}
```

```
> hash["ewha"] = "hack"  
hash["like"] = "your"  
hash["lion"] = "life"  
hash["hello"] = "hi"
```

자동으로 부여된 숫자대신!

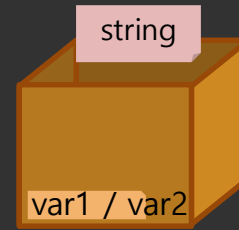
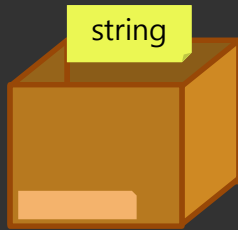
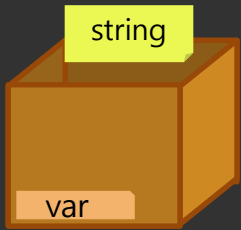
원하는 이름으로 index를 지정한다.



LIKE LION

심볼 (Symbol)

박스의 위치 = **object_id** (주소공간)



```
var = "string"  
var.object_id != "string".object_id
```

```
>>> 18032800  
      18022200
```

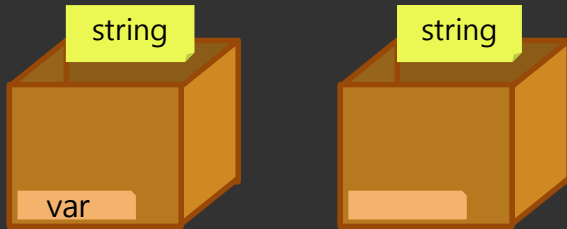
```
var1 = :string  
var2 = :string  
var1.object_id == var2.object_id
```

```
>>> 802268  
      802268
```

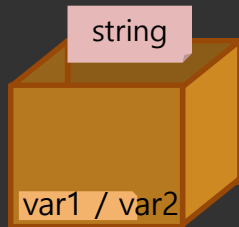


LIKE LION

심볼 (Symbol)



Hash의 Key로 Symbol을 사용한다.



- > 공간의 낭비X
- > 속도도 빠름
- > 재사용 O



LIKE LION

조건문 (control flow)

if (조건문1)

block1



조건문1이 true이면 block1 실행

elsif (조건문2)

block2



조건문1이 false이고 조건문2가 true이면
block2 실행

else

block3



조건문1, 조건문2 모두 false이면 block3 실행

end



LIKE LION

시험 점수를 입력받아 90 ~ 100점은 A, 80 ~ 89점은 B, 70 ~ 79점은 C, 60 ~ 69점은 D, 나머지 점수는 F를 출력하는 프로그램을 작성하시오.

첫째 줄에 시험 점수가 주어진다. 시험 점수는 0보다 크거나 같고, 100보다 작거나 같은 자연수이다.

```
puts "0보다 크거나 같고, 100보다 작거나 같은 자연수를 입력해주세요"  
test = gets.to_i
```

예제 입력 1 복사

100

예제 출력 1 복사

A



LIKE LION

시험 점수를 입력받아 90 ~ 100점은 A, 80 ~ 89점은 B, 70 ~ 79점은 C, 60 ~ 69점은 D, 나머지 점수는 F를 출력하는 프로그램을 작성하시오.

첫째 줄에 시험 점수가 주어진다. 시험 점수는 0보다 크거나 같고, 100보다 작거나 같은 자연수이다.

```
puts "0보다 크거나 같고, 100보다 작거나 같은 자연수를 입력해주세요"
test = gets.to_i
if test>=90 && test<=100
  puts "A"
elsif test>=80 && test<=89
  puts "B"
elsif test>=70 && test<=79
  puts "C"
elsif test>=60 && test<=69
  puts "D"
else
  puts "F"
end
```

```
0보다 크거나 같고, 100보다 작거나 같은 자연수를 입력해주세요
97
A
```

예제 입력 1 복사

100

예제 출력 1 복사

A



LIKE LION

반복문 (Loop)

for loop

루프를 반복횟수를 아는 경우

```
for var in range
```

```
    block
```

```
end
```

1...10 : 1~9의 값 (배제)
1..10 : 1~10의 값 (포함)

while loop

루프를 반복횟수를 모르는 경우
(특정 조건이 참일 때까지!)

```
count = 1
```

```
While (조건)
```

```
    block
```

```
    count += 1
```

```
end
```



LIKE LION

N을 입력받은 뒤, 구구단 N단을 출력하는 프로그램을 작성하시오. 출력 형식에 맞춰서 출력하면 된다.

첫째 줄에 N이 주어진다. N은 1보다 크거나 같고, 9보다 작거나 같다.

```
puts "구구단 몇단?"  
n = gets.to_i
```

예제 입력 1 복사

2

예제 출력 1 복사

```
2 * 1 = 2  
2 * 2 = 4  
2 * 3 = 6  
2 * 4 = 8  
2 * 5 = 10  
2 * 6 = 12  
2 * 7 = 14  
2 * 8 = 16  
2 * 9 = 18
```



LIKE LION

N을 입력받은 뒤, 구구단 N단을 출력하는 프로그램을 작성하시오. 출력 형식에 맞춰서 출력하면 된다.

첫째 줄에 N이 주어진다. N은 1보다 크거나 같고, 9보다 작거나 같다.

```
puts "구구단 몇단?"
n = gets.to_i
for i in 1..9
  puts "#{n} * #{i} = #{n*i} "
end
```

구구단 몇단?

```
9
9 * 1 = 9
9 * 2 = 18
9 * 3 = 27
9 * 4 = 36
9 * 5 = 45
9 * 6 = 54
9 * 7 = 63
9 * 8 = 72
9 * 9 = 81
```

예제 입력 1 복사

2

예제 출력 1 복사

```
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
```



LIKE LION

메소드 (Method)

1.Times

"hello".reverse

arr.each

...

"."로 소환



LIKE LION

.each

```
array = {"a", "b", "c"}  
array.each do |x|  
  puts "#{x}"  
end
```

[배열에서]
한번에 하나씩 각 요소에 식을 적용
||사이에 원하는 변수설정 (해당 loop에서만)

```
array = {"a"=>1, "b"=>2,  
"c"=>3}  
array.each do |x, y|  
  puts "#{x} : #{y}"  
end
```

[해시에서]
한번에 한 섹션을 반복하기 때문에,
키 - 값 쌍을 반복 할 수 있다.
x에 key, y에 value가 온다.



LIKE LION


```
likelion=["hack", "your", "likf"]
ewha = {
  "가건" => "가족 건강과 간호",
  "컴프실" => "컴퓨터 프로그래밍 실습",
  "서미이" => "서양 미술의 이해",
  "나리" => "나눔 리더십"
}

likelion.each{|x| puts "#{x}"}
ewha.each{|x, y| puts "#{x} : #{y}"}
```

```
hack
your
likf
가건 : 가족 건강과 간호
컴프실 : 컴퓨터 프로그래밍 실습
서미이 : 서양 미술의 이해
나리 : 나눔 리더십
```



LIKE LION

```
likelion=["hack", "your", "likf"]
ewha = {
  "가건" => "가족 건강과 간호",
  "컴프실" => "컴퓨터 프로그래밍 실습",
  "서미이" => "서양 미술의 이해",
  "나리" => "나눔 리더십"
}

likelion.each{|x| puts "#{x}"}
ewha.each{|x, y| puts "#{x} : #{y}"}
```

```
likelion.each{puts 3}
```

```
3
3
3
```

```
hack
your
likf
가건 : 가족 건강과 간호
컴프실 : 컴퓨터 프로그래밍 실습
서미이 : 서양 미술의 이해
나리 : 나눔 리더십
```

반복문처럼 쓸 수 있다.

그렇지만 이럴 경우

.length

for loop



LIKE LION

.rand

rand

```
>> 0.19844698345
```

rand(1000)

```
>> 668
```

rand(5.0...9.0)

```
>> 6.489320
```

```
array = ["ㄱ", "ㄴ", "ㄷ"]
```

array.rand

```
>> "ㄷ"
```

그냥 rand만 써서 무작위로 숫자를 뽑을 수 있다.

rand : 0~1사이의 실수

()안에 숫자를 써서 해당 범위내의 숫자를 뽑는다.

rand(x) : 0~x사이의 정수

범위 지정을 위해 ..과 ...을 쓸 수 있다.

숫자를 실수로(소수점이 있는 수) 쓸 수 있다.

메소드로 사용해서 배열, 해시에 있는 값을

랜덤하게 뽑을 수 있다.



LIKE LION

.sort_by

배열을 지정한 목적에 맞는 순서로 정렬

```
colors = {"blue"=> 3, "green"=>1, "red"=>2}
```

```
colors = colors.sort_by do |color, count|
```

```
  count
```

```
end
```

|color, count|에서 count로 지정을 했기 때문에 count기준
즉, 숫자기준으로 정렬한다. (오름차순 정렬)

```
>>> [{"green", 1}, {"red", 2}, {"blue", 3}]
```

Q. 여기서 sort_by대신 sort를 쓴다면?



LIKE LION

.select

특정 기준을 충족하는 값에 대해 해시를 필터링한다.

자신이 설정한 **조건과 일치**하는 키-값 쌍을 **선택**

```
grades = { hack: 100, your: 95, life :97, coding: 92}
```

```
grades.select{ |name, grade| grade < 97}
```

```
>> {:your => 95, :coding => 92}
```

```
Grades.select{ |k,v| k == :hack}
```

```
>> {:alice => 100}
```



LIKE LION

.length

문자열, 배열의 길이를 알려준다.

●색은 대체가능

.times

초소형 for loop , 지정된 횟수만큼 반복한다.

.reverse

문자열, 배열의 역방향 버전으로 만든다.

.upcase / .downcase

문자열을 모두 대문자 / 소문자로 변환한다.

.chomp

여분의 줄을 제거한다 (gets랑 많이 쓰인다.)

.capitalize

문자열의 가장 첫번째를 대문자로 변환한다.

.include? "char"

문자열에 "char"가 들어갔는가? (true / false)

gsub(/char/, "replace")

문자열 "char"을 "replace"로 값을 바꾼다.

.split(",")

","를 볼때 마다 문자열을 분할하여 배열로 만든다.

Hash.new("nothing")

초기값이 "nothing"인 해시를 생성한다.

.to_s / to_sym

심볼을 문자열로 / 문자열을 심볼로 변환한다.

.intern

.to_sym과 같은 의미이다



LIKE LION

실 습



LIKE LION

EWHA 교과목 줄임표



LIKE LION

EWHA 교과목 줄임표

교과목 줄임표.txt - 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

사고의 변형", "경사"=>"경영 사회학", "경원"=>"경제 원론", "경통"=>"경영 통계학", "법제"=>"법과 제도", "고글"=>"고전읽기 와 글쓰기", "고영"=>"고급 영어", "관미"=>"관계의 미학", "교합"=>"교양 합성", "국개합"=>"국제 구호와 개발협력", "글시아"=>"글로벌시대 아시아 여성", "글커개"=>"글로벌 커리어 개발과 기업가 정신", "기세"=>"기독교와 세계", "기작기"=>"기사 작성 기초", "김미시"=>"김교수님 미시경제학", "심이"=>"심리학의 이해", "나리"=>"나눔더샵", "뇌기초"=>"뇌인식과학 기초", "뇌주발"=>"뇌인지과학 주제발견", "뇌주탐"=>"뇌인지과학 주제탐구", "동고이"=>"동양고전의 현대적 이해", "동문문"=>"동아시아 (동양) 문학으로 문화 읽기", "동미이"=>"동양 미술의 이해", "동서인"=>"동서양의 인간이해", "동역부"=>"동아시아 역사 부쟁과 한국사", "대우영"=>"대학생을 위한 운동과 영양", "한여사"=>"한국 여성사", "한정"=>"한정 정치", "한현삼"=>"한국 현대사와 삶 읽기", "디네검"=>"디지털 네이티브와 검색어친", "디동문"=>"디지털미디어 시대와 동아시아 문자", "디스이"=>"디지털 스토리 텔링의 이해", "디매예"=>"디지털 매체와 예술", "디스개"=>"디지털 스토리 텔링 개론", "마과"=>"마케팅 과론", "모애플"=>"모바일 앱 프로그래밍", "무이"=>"무용의 이해", "문경영"=>"문화 예술 경영", "무매이"=>"문학으로 이해하는 매체와 이야기", "문사역"=>"문학으로 보는 사랑의 역사", "문삼"=>"문학과 삶", "문영"=>"문학과 영화", "문예르"=>"문학과 예술을 통해 본 르네상스 역사", "문현세"=>"문화로 읽는 현대사의 세계", "미미"=>"미학과 미술", "미시"=>"미시 경제학", "미환이"=>"미래를 위한 환경의 이해", "방미시"=>"방교수님 미시경제학", "북문생"=>"북한 주민의 문화와 생활", "북정이"=>"북한 정치의 이해", "간후"=>"간호와 후버니즘", "브스럼"=>"브라운 스펙트럼", "사과고"=>"사회과학 고전읽기", "사운"=>"사랑과 윤리", "사에"=>"사물과 예술", "사인개"=>"사물인터넷개론", "사픽이"=>"사이언스 픽션의 이해", "사학이"=>"사회학의 이해", "서미이"=>"서양 미술의 이해", "서음이"=>"서양 음악의 이해", "성문연"=>"성문화 연구", "성사"=>"성의 사회학", "세문예"=>"세계와 문화예술", "생윤법"=>"생명 윤리와 법", "시경세"=>"시장 경제 세미나", "시문미"=>"시각 문화로서 미술 작품 읽기", "시문트"=>"시각적 사고를 통한 일상의 문화와 트렌드", "시법"=>"시민 생활과 법", "신상문"=>"신화 상상력 문화", "스글"=>"스토리 텔링과 글쓰기", "스세"=>"스포츠와 세계화", "여사정"=>"여성과학자의 이해", "동철이"=>"동양 철학의 이해", "동행"=>"동물의 행동", "여예"=>"여성과학의 이해", "여지"=>"여행과 지리", "예그시"=>"예술가와 그의 시대", "예사원"=>"예술의 사회화적 읽기", "오이감"=>"오페라의 이해와 감상", "외정이"=>"외교정책의 이해", "외정분"=>"외교정책의 분석", "왜츨책"=>"인간은 왜 춤을 추는가", "우글"=>"우리말과 글쓰기", "움감교"=>"움직임을 통한 감성교양", "위판심"=>"위험판단 심리학", "음과기"=>"음악과 과학기술", "응동입"=>"응용 통계 입문", "임미심"=>"인터랙티브 미디어 심리학", "인발가"=>"인간 발달과 가족", "인법"=>"인터넷과 법", "이십이"=>"인간의 심리학적 이해", "인철사"=>"인간과 철학적 사유", "인한사"=>"인물로 읽는 한국사", "임물"=>"일반 물리학", "일생"=>"일반 생물학", "일화"=>"일반 화학", "정개"=>"정치학개론", "정미"=>"정보기술과 미래사회", "정통"=>"정보화 사회와 통계", "주서근"=>"주제로 살펴보는 서양 근현대사", "중역문"=>"중국의 역사와 문화", "점판"=>"재무관리", "죄범"=>"죄와 벌", "지역"=>"지구의 역사", "질문역"=>"질병과 몸의 역사", "창상기"=>"창조와 상상의 기술", "청자이"=>"청춘의 자기 이해", "철은"=>"철학과 은유들", "커사"=>"커뮤니케이션과 사회", "연이실"=>"연극의 이론과 실제", "영이젠"=>"영상 이미지와 제도", "커탐"=>"커리어 탐색과 이해", "컴마"=>"컴퓨터와 마음", "컴사프"=>"컴퓨팅적 사고와 프로그래밍", "컴프실"=>"컴퓨터 프로그래밍 실습", "키중문"=>"키워드론 읽는 중국 문화", "패전비"=>"패러다임 전환과 비판 이론", "포모철"=>"포스트 모더니즘과 철학", "약건"=>"약과 건강", "여몽창"=>"여성의 몸과 창조적 움직임", "한새이"=>"한국사의 새로운 이해", "한미이"=>"한국 미술의 이해", "한문세"=>"한상 문학의 세계", "한재국"=>"환경재난과 국제협력", "현미사"=>"현대 미술사", "인문세"=>"인문학으로 세상읽기", "현한정"=>"현대 한국 정치", "호모"=>"호모엠포터쿠스와 인간의 이해", "인언"=>"인간과 언어", "회경원"=>"회계 정보 원리", "릿사"=>"likelion"

중구 난방이야!!! ruby를 이용해서 정리해보자!!



LIKE LION

script.rb

```
1 ewha = {
2   "가건">"가족 건강과 간호", "거식">"건강한 식생활", "현문기">"현대 문화와 기독교", "현물인">"현대 물리학과 인간 사고의 변혁", "경사">"경영 사회학", "경원">"경제 원론", "경통">"경영 통계학", "법전">"법과 전더", "고글">"고전읽기와 글쓰기", "고영">"고급 영어", "관미">"관계의 미학", "교합">"교양 합창", "국개협">"국제 구조와 개발협력", "글시아">"글로벌시대 아시아 여성", "글커개">"글로벌 커리어 개발과 기업가 정신", "기세">"기독교와 세계", "기작기">"기사 작성 기초", "김미시">"김교수님 미시경제학", "실이">"실리학의 이해", "나리">"나눔리더십", "뇌기초">"뇌인지과학 기초", "뇌주발">"뇌인지과학 주제발견", "뇌주탐">"뇌인지과학 주제탐구", "동고이">"동양고전의 현대적 이해", "동문문">"동아시아 (동양) 문학으로 문화 읽기", "동미미">"동양 미술의 이해", "동서인">"동서양의 인간이해", "동역문">"동아시아 역사 문명과 한국사", "대문영">"대학생을 위한 문학과 영양", "한여사">"한국 여성사", "한정">"한국 정치", "한현삼">"한국 현대사와 삶 읽기", "디네검">"디지털 네이티브와 검색엔진", "디동문">"디지털미디어 시대의 동아시아 문자", "디스이">"디지털 스토리텔링의 이해", "디매예">"디지털 매체와 예술", "디스개">"디지털 스토리텔링 개론", "마관">"마케팅 관리", "모애플">"모바일 앱 프로그래밍", "무이">"무용의 이해", "문경열">"문화 예술 경영", "문매이">"문학으로 이해하는 매체와 이야기", "문사역">"문학으로 보는 사랑의 역사", "문삼">"문학과 삶", "문영">"문학과 영화", "문메르">"문화와 예술을 통해 본 르네상스 역사", "문현세">"문화로 읽는 현대사의 세계", "미미">"미학과 미술", "미시">"미시 경제학", "미환이">"미래를 위한 환경의 이해", "밤미시">"밤교수님 미시경제학", "복문생">"북한 주민의 문화와 생활", "북정이">"북한 정치의 이해", "간혹">"간호와 휴머니즘", "보스럼">"브레인 스펙트럼", "사과고">"사회과학 고전읽기", "사문">"사랑과 윤리", "사에">"사물과 예술", "사인개">"사물인터넷개론", "사피이">"사이언스 픽션의 이해", "사학이">"사회학의 이해", "서미이">"서양 미술의 이해", "서음이">"서양 음악의 이해", "성문연">"성 문화 연구", "성사">"성의 사회학", "세문예">"세계와 문화예술", "생윤법">"생명 윤리와 법", "시경세">"시장 경제 세미나", "시문미">"시각 문화로서 미술 작품 읽기", "시문트">"시각적 사고를 통한 일상의 문화와 트렌드", "시법">"시민 생활과 법", "신상문">"신화 상상력 문화", "스글">"스토리텔링과 글쓰기", "스세">"스포츠와 세계화", "여사정">"여성과 사회정의", "동철이">"동양 철학의 이해", "동현">"동물의 행동", "여예">"여성과 예술", "여지">"여행과 지리", "에고시">"예술가와 그의 시대", "메사왕">"예술의 사회학적 읽기", "오이감">"오페라의 이해와 감상", "외정이">"외교정책의 이해", "외정문">"외교정책의 분석", "왜츄츄">"인간은 왜 춤을 추는가", "우글">"우리의말과 글쓰기", "움감묘">"움직임을 통한 감정표현", "위판신">"위험판단 심리학", "음과기">"음악과 과학 기술", "음통인">"음용 통계 입문", "인미심">"인터랙티브 미디어 심리학", "인발가">"인간 발달과 가족", "인번">"인터넷과 법", "인심이">"인간의 심리학적 이해", "인철사">"인간과 철학적 사유", "인한사">"인물로 읽는 한국사", "일물">"일반 물리학", "일생">"일반 생물학", "일화">"일반 화학", "정개">"정치학개론", "정미">"정보기술과 미래 사회", "정통">"정보화 사회와 통계", "주서근">"주제로 살펴보는 서양 근현대사", "중역문">"중국의 역사와 문화", "절관">"재무관리", "죄벌">"죄와 벌", "지역">"지구의 역사", "질문역">"질병과 몸의 역사", "창상기">"창조와 상상의 기술", "청자이">"청춘의 자기 이해", "철은">"철학과 은유들", "커사">"커뮤니케이션과 사회", "연이실">"연극의 이론과 실제", "영이첸">"영상 이미지와 전더", "커탐">"커리어 탐색과 이해", "컴마">"컴퓨터와 마음", "컴사프">"컴퓨팅적 사고와 프로그래밍", "컴프실">"컴퓨터 프로그래밍 실습", "키중문">"키워드로 읽는 중국 문화", "패전비">"패러다임 전환과 비판 이론", "포모철">"포스트 모더니즘과 철학", "약건">"약과 건강", "여몽창">"여성의 몸과 창조적 움직임", "한새이">"한국사의 새로운 이해", "한미이">"한국 미술의 이해", "한문세">"한상 문학의 세계", "한재국">"한 경제난과 국제협력", "현미사">"현대 미술사", "인문세">"인문학으로 세상읽기", "현한정">"현대 한국 정치", "호모">"호모헵타투루스와 인간의 이해", "인언">"인간과 언어", "회정원">"회계 정보 원리", "멋사">"like lion"
```

Codecademy에 옮기고! 이제 코딩을 해보자!



LIKE LION

다행히도 hash형태로 정렬이 되어있다!
Ruby의 각종 메소드, 문법을 사용해보자!



LIKE LION

1. ㄱㄴㄷ 순에 맞게 정렬한 후 콘솔에 출력하기

가건:가족 건강과 간호
간휴:간호와 휴머니즘
건식:건강한 식생활
경사:경영 사회학
경원:경제 원론
경통:경영 통계학
고을:고전읽기와 글쓰기
고영:고급 영어
관미:관계의 미학
교한:교양 학창
국개협:국제 구호와 개발협력
글시아:글로벌시대 아시아 여성
글커개:글로벌 커리어 개발과 기업가 정신
기세:기독교와 세계
기작기:기사 작성 기초
김미사:김교수님 미시경제학
나리:나눔리더십
뇌기초:뇌인지과학 기초
뇌주발:뇌인지과학 주제발견
뇌주말:뇌인지과학 주제탐구
대문영:대학생을 위한 문명과 영양
동고이:동양고전의 현대적 이해
동문문:동아시아 (동양) 문학으로 문화 읽기
동미미:동양 미술의 이해
동서인:동서양의 인간이해
동역문:동아시아 역사 문명과 한국사
동철이:동양 철학의 이해
동행:동물의 행동
디네검:디지털 네이티브와 검색엔진
디동문:디지털미디어 시대의 동아시아 문자
디대예:디지털 매체와 예술
디스개:디지털 스토리 텔링 개론
디스아:디지털 스토리 텔링의 이해
마관:마케팅 관리
멋사:likelion
모애플:모바일 앱 프로그래밍
무이:무용의 이해
문경영:문화 예술 경영
문대이:문학으로 이해하는 매체와 이야기
문사역:문학으로 보는 사상의 역사
문삶:문학과 삶
문영:문학과 영화
문예르:문화와 예술을 통해 본 르네상스 역사
문현세:문화로 읽는 현대사의 세계
미미:미학과 미술
미시:미시 경제학
미환이:미래를 위한 환경의 이해
방미사:방교수님 미시경제학
반젠:법과 젠더
북문생:북한 주민의 문화와 생활

북절이:북한 정치의 이해
브스컬:브레인 스펙트럼
사과고:사회과학 고전읽기
사예:사물과 예술
사운:사상과 윤리
사인개:사물인터넷개론
사픽이:사이언스 픽션의 이해
사학이:사회학의 이해
생문법:생명 윤리와 법
서미이:서양 미술의 이해
서음이:서양 음악의 이해
성문연:성 문화 연구
성사:성의 사회학
세문예:세계와 문화예술
스글:스토리 텔링과 글쓰기
스세:스포츠와 사회학
시경제:시장 경제 세미나
시문미:시각 문화로서 미술 작품 읽기
시문트:시각적 사고를 통한 일상의 문화와 트렌드
시반:시민 생활과 법
신상문:신화 상상력 문화
심이:심리학의 이해
약간:약과 건강
여문창:여성의 몸과 창조적 움직임
여사절:여성과 사회정의
여예:여성과 예술
여지:여학과 지리
연이상:연극의 이론과 실제
영이젠:영상 이미지와 젠더
예그시:예술가와 그의 시대
예사왕:예술의 사회학적 읽기
오이강:오페라의 이해와 감상
왜총책:인간은 왜 총을 추는가
외정문:외교정책의 분석
외정이:외교정책의 이해
우글:우리말과 글쓰기
움감고:움직임을 통한 감정조형
위판심:위험판단 심리학
음과기:음악과 과학 기술
음통인:음용 통계 인문
인문세:인문학으로 세상읽기
인미심:인터랙티브 미디어 심리학
인발가:인간 발달과 가족
인반:인터넷과 법
인심이:인간의 심리학적 이해
인언:인간과 언어
인철사:인간과 철학적 사유
인한사:인물로 읽는 한국사
일물:일반 물리학
일생:일반 생물학

일화:일반 화학
절관:재무관리
절개:정치학개론
절미:정보기술과 미래사회
정통:정보화 사회와 통계
죄벌:죄와 벌
주서근:주제로 살펴보는 서양 근현대사
중역문:중국의 역사와 문화
지역:지구의 역사
질문역:질병과 몸의 역사
창상기:창조와 상상의 기술
철은:철학과 은유들
참자이:청춘의 자기 이해
커사:커뮤니케이션과 사회
커탐:커리어 탐색과 이해
컴마:컴퓨터와 마음
컴사프:컴퓨팅적 사고와 프로그래밍
컴프실:컴퓨터 프로그래밍 실습
키중문:키워드로 읽는 중국 문화
패전비:패러다임 전환과 비판 이론
포모철:포스트 모더니즘과 철학
한미이:한국 미술의 이해
한새이:한국사의 새로운 이해
한여사:한국 여성사
한정:한국 정치
한현삼:한국 현대사와 삶 읽기
현문기:현대 문화와 기독교
현물인:현대 물리학과 인간 사고의 변혁
현미사:현대 미술사
현한정:현대 한국 정치
호모:호모헹파티쿠스와 인간의 이해
환문세:환상 문학의 세계
환재국:환경재난과 국제협력
회정원:회계 정보 원리



LIKE LION

2. 졸업말표에 정리된 과목 개수 출력하기

졸업말표에 있는 과목은 총 134개 입니다.

여기 주의!



LIKE LION

3. 줄임말 + 과목 쌍 찾기

사용자 입력!

찾고 싶은 줄임말을 입력하세요

호모

[["호모", "호모엠펙티쿠스와 인간의 이해"]]

메소드? Hash호출 어떤 걸 쓰지?

변수인 ewha가 어떻게 자료가 정렬되었는지 보자



LIKE LION

3. 줄임말 + 과목 쌍 찾기

사용자 입력!

찾고 싶은 줄임말을 입력하세요

호모

[["호모", "호모엠평티쿠스와 인간의 이해"]]

메소드? Hash호출 어떤 걸 쓰지?

변수인 ewha가 어떻게 자료가 정렬되었는지 보자

포모철 / 여인실 /디네검 /환재국 / 컴프실

5개 쌍을 찾아서 console에 출력하기!



LIKE LION

Solution



LIKE LION

1. ↑↓ 순에 맞게 정렬한 후 콘솔에 출력하기

```
ewha = ewha.sort_by{|key, value| key}
ewha.each{|key, value| puts "#{key}:#{value}"}
```

줄임표인 key값을 기준으로 **.sort_by** 메소드를 사용!

each를 통해 각 키-값 쌍을 받아온다.

key : value 형태로 console에 출력한다.



LIKE LION

2. 졸업말표에 정리된 과목 개수 출력하기

졸업말표에 있는 과목은 총 134개 입니다.

```
puts " "  
puts "졸업말표에 있는 과목은 총 #{ewha.length}개 입니다."
```

length를 사용해서 변수 ewha의 키-값 쌍의 개수를 찾는다.
string 안에 변수넣는 방법 #{ } 잊지 말자!



LIKE LION

sort_by를 한 후에 ewha를 출력해 보셨나요?

```
ewha = ewha.sort_by{|key, value| key}  
puts ewha
```



LIKE LION

해시에서 2중 배열의 형태로?!

[["가건", "가족 건강과 간호"], ["간휴", "간호와 휴머니즘"], ["건식", "건강한 식생활"], ["경사", "경영 사회학"], ["경영", "경제 원론"], ["경통", "경영 통계학"], ["고글", "고전읽기와 글쓰기"], ["고영", "고급 영어"], ["관미", "관계의 미학"], ["교한", "교양 함양"], ["국개현", "국제 구호와 개발협력"], ["글시아", "글로벌 시대 아시아 여성"], ["글커개", "글로벌 커리어 개발과 기업가 정신"], ["기세", "기독교와 세계"], ["가작기", "기사 작성 기초"], ["김미시", "김교수님 미시경제학"], ["나리", "나눔리더십"], ["뇌기초", "뇌인지과학 기초"], ["뇌주발", "뇌인지과학 주제발견"], ["뇌주탐", "뇌인지과학 주제탐구"], ["대운영", "대학생을 위한 운동과 영양"], ["동고이", "동양고전의 현대적 이해"], ["동문문", "동아시아 (동양) 문학으로 문화 읽기"], ["동미이", "동양 미술의 이해"], ["동서인", "동서양의 인간이해"], ["동역분", "동아시아 역사 분쟁과 한국사"], ["동철이", "동양 철학의 이해"], ["동행", "동물의 행동"], ["디네검", "디지털 네이티브와 검색엔진"], ["디동문", "디지털미디어 시대의 동아시아 문자"], ["디매예", "디지털 매체와 예술"], ["디스개", "디지털 스토리텔링 개론"], ["디스이", "디지털 스토리텔링의 이해"], ["마관", "마케팅 관리"], ["멋사", "Likelion"], ["모애플", "모바일 앱 프로그래밍"], ["무이", "무용의 이해"], ["문경영", "문화 예술 경영"], ["문매이", "문학으로 이해하는 매체와 이야기"], ["문사역", "문학으로 보는 사람의 역사"], ["문삶", "문화와 삶"], ["문영", "문화와 영화"], ["문예르", "문화와 예술을 통해 본 르네상스 역사"], ["문헌세", "문화로 읽는 현대시의 세계"], ["미미", "미학과 미술"], ["미시", "미시 경제학"], ["미환이", "미래를 위한 환경의 이해"], ["발미시", "방교수님 미시경제학"], ["법젠", "법과 젠더"], ["북문생", "북한 주민의 문화와 생활"], ["북정이", "북한 정치의 이해"], ["브스럼", "브레인 스펙트럼"], ["사과고", "사회과학 고전읽기"], ["사예", "사물과 예술"], ["사윤", "사랑과 윤리"], ["사인개", "사물인터넷개론"], ["사픽이", "사이언스 픽션의 이해"], ["사학이", "사회학의 이해"], ["생윤법", "생명 윤리와 법"], ["서미이", "서양 미술의 이해"], ["서음이", "서양 음악의 이해"], ["성문연", "성 문화 연구"], ["성사", "성의 사회학"], ["세문예", "세계와 문화예술"], ["스글", "스토리텔링과 글쓰기"], ["스세", "스포츠와 세계화"], ["시경세", "시장 경제 세미나"], ["시문미", "시각 문화로서 미술 작품 읽기"], ["시문트", "시각적 사고를 통한 일상의 문화와 트렌드"], ["시법", "시민 생활과 법"], ["신상문", "신화 상상력 문화"], ["심이", "심리학의 이해"], ["악건", "악과 건강"], ["여몸창", "여성의 몸과 창조적 움직임"], ["여사절", "여성과 사회정의"], ["여예", "여성과 예술"], ["여지", "여행과 지리"], ["연이실", "영국의 이론과 실제"], ["영이젠", "영상 이미지와 젠더"], ["예그시", "예술가와 그의 시대"], ["예사읽", "예술의 사회학적 읽기"], ["오이감", "오페라의 이해와 감상"], ["왜츨취", "인간은 왜 춤을 추는가"], ["외정분", "외교정책의 분석"], ["외정이", "외교정책의 이해"], ["무글", "우리말과 글쓰기"], ["움감교", "움직임을 통한 감정교침"], ["위판심", "위험판단 심리학"], ["음과기", "음악과 과학 기술"], ["음통입", "음용 통계 입문"], ["인문세", "인문학으로 세상읽기"], ["인미심", "인터넷티브 미디어 심리학"], ["인발가", "인간 발달과 가족"], ["인법", "인터넷과 법"], ["인심이", "인간의 심리학적 이해"], ["인언", "인간과 언어"], ["인철사", "인간과 철학적 사유"], ["인한사", "인물로 읽는 한국사"], ["일물", "일반 물리학"], ["일생", "일반 생물학"], ["일화", "일반 화학"], ["젠관", "재무관리"], ["정개", "정치학개론"], ["정미", "정보기술과 미래사회"], ["절물", "정문화 사회와 통계"], ["죄벌", "죄와 벌"], ["주서근", "주제로 살펴보는 서양 근현대사"], ["중역문", "중국의 역사와 문화"], ["지역", "지구의 역사"], ["질문역", "질병과 몸의 역사"], ["창상기", "창조와 상상의 기술"], ["철", "철학과 은유들"], ["청자이", "청춘의 자기 이해"], ["커사", "커뮤니케이션과 사회"], ["커탐", "커리어 탐색과 이해"], ["컴마", "컴퓨터와 마음"], ["컴사프", "컴퓨팅적 사고와 프로그래밍"], ["컴프실", "컴퓨터 프로그래밍 실습"], ["키중문", "키워드 읽는 중국 문화"], ["패전비", "패러다임 전환과 비판 이론"], ["포모철", "포스트 모더니즘과 철학"], ["한미이", "한국 미술의 이해"], ["한새이", "한국사의 새로운 이해"], ["한여사", "한국 여성사"], ["한정", "한국 정치"], ["한현삶", "한국 현대시와 삶 읽기"], ["현문기", "현대 문화와 기독교"], ["현물인", "현대 물리학과 인간 사고의 변화"], ["현미사", "현대 미술사"], ["현한절", "현대 한국 정치"], ["호모", "호모헤파티쿠스와 인간의 이해"], ["환문세", "환상 문학의 세계"], ["환재국", "환경재난과 국제협력"], ["회정원", "회계 정보 원리"]]



LIKE LION

3. 줄임말 + 과목 쌍 찾기

```
puts ""  
puts "찾고 싶은 줄임말을 입력하세요"  
name = gets.chomp  
puts ewha.select{|key, value| key=="#{name}"}
```

gets.chomp를 이용해서 사용자로부터 입력받아 name저장!
.select 메소드로 name과 일치하는 key값(줄임말)을 찾는다.



LIKE LION

3. 줄임말 + 과목 쌍 찾기

```
puts ""  
puts "찾고 싶은 줄임말을 입력하세요"  
name = gets.chomp  
puts ewha.select{|key, value| key=="#{name}"}
```

왜 "#{name}"을 사용하나요?

Key 값이 string 값으로 되어있어서, name도 string으로!

puts name하면? ""없이 그냥 문자만 있는 값이 나온다!



LIKE LION

루비

codecademy

내용 정리

https://github.com/mjung1798/ruby_language/blob/master/%EB%A3%A8%EB%B9%84.pdf



LIKE LION



이 외에 모든 ruby 문법 정리

<https://ruby-doc.org/core-2.2.0/index.html#methods>



LIKE LION

Part 2 . Web & Rails



LIKE LION

HTML 파일

= 웹 문서



LIKE LION

HTML



LIKE LION

HTML



LIKE LION

HTML

```
<!DOCTYPE html>
<html lang="en">
  <head>
    .
  </head>
  <body>
    <a href="">..</a>
  </body>
</html>
```



LIKE LION

N 쇼핑

세상에 하나뿐인 핸드메이드는 **리빙원도** 로!



[연합뉴스](#) > 안희정 영장심사 불출석...법원, 28일 오후로 기일 다시 잡아

[네이버뉴스](#) [연예](#) [스포츠](#) [경제](#) [랭킹](#)

[뉴스스탠드](#) > [전체 언론사](#) | [MY 뉴스](#)



| | | | | | |
|------------------------|-------------------------|----------------------|---|----------------------------------|---------------------------|
| 연합뉴스TV | 석간 문화일보 | 세계일보 | 디지털타임스 | The Korea Herald | 노컷뉴스 |
| 국민일보 | 서울경제 | 한국경제 | 구독 기사보기 | 서울신문 | ChosunBiz |
| Forbes | 티브이데일리 | 일요신문 | SBS CNBC | Da 디지털데일리 | THE ASIAN |

[자동차](#) [패션뷰티](#) [맘·키즈](#) [건강](#) [웹툰](#) [게임](#) [TV연예](#) [뮤직](#) [영화](#) [책문화](#) [함께N](#)

건강



입맛없다 퇴근하면 폭식...
'야식증후군' 아니세요?

| | | |
|------|------------|----------------|
| 아이디 | 로그인 | IP보안 ON |
| 비밀번호 | | 일회용 로그인 |

☐ 로그인 상태 유지 ☐ 회원가입 [아이디/비밀번호 찾기](#)

03.26. (월) 뉴스 2/6 < >

[TV뉴스](#) **LIVE** 저녁 메인뉴스 보기

[학습](#) 최신 시사상식 [영어](#)로 듣는 뉴스 [영어](#) 특특
[부동산](#) 재건축 떠난 돈 재개발로 몰려...1분기 입주...



나만의 파우치 만들기!

내 취향이 담겨서 그런가?
보면 볼수록 더 예뻐!
[구매하기](#) >

AD 네이버 리빙원도

[쇼핑](#) [상품](#) [쇼핑몰](#) [MEN](#)

G마켓 · 옥션 · 11번가 · 위메프 · 티몬 · 이마트몰 · 신세계몰
GS샵 · 롯데몰 · 롯데닷컴 · CJmall · 현대H몰 · 올리브영



LIKE LION

그렇지만
Html로만 웹서비스를?



LIKE LION

N 쇼핑

세상에 하나뿐인 핸드메이드는 **리빙원도** 로!



[연합뉴스](#) > 안희정 영장심사 불출석...법원, 28일 오후로 기일 다시 잡아

[네이버뉴스](#) [연예](#) [스포츠](#) [경제](#) [랭킹](#)

[뉴스스탠드](#) > [전체 언론사](#) | [MY 뉴스](#)



| | | | | | |
|------------------------|-------------------------|----------------------|---|----------------------------------|---------------------------|
| 연합뉴스TV | 석간 문화일보 | 세계일보 | 디지털타임스 | The Korea Herald | 노컷뉴스 |
| 국민일보 | 서울경제 | 한국경제 | 구독 기사보기 | 서울신문 | ChosunBiz |
| Forbes | 티브이데일리 | 일요신문 | SBS CNBC | Da 디지털데일리 | THE ASIAN |

[자동차](#) [패션뷰티](#) [맘·키즈](#) [건강](#) [웹툰](#) [게임](#) [TV연예](#) [뮤직](#) [영화](#) [책문화](#) [함께N](#)

건강



입맛없다 퇴근하면 폭식...
'야식증후군' 아니세요?

| | | |
|------|------------|----------------|
| 아이디 | 로그인 | IP보안 ON |
| 비밀번호 | | 일회용 로그인 |

☐ 로그인 상태 유지 ☐ 회원가입 [아이디/비밀번호 찾기](#)

03.26. (월) 뉴스 2/6 < >

[TV뉴스](#) **LIVE** 저녁 메인뉴스 보기

[학습](#) 최신 시사상식 [영어](#)로 듣는 뉴스 [영어](#) 특목

[부동산](#) 재건축 떠난 돈 재개발로 몰려...1분기 입주...



나만의 파우치 만들기!

내 취향이 담겨서 그런가?
보면 볼수록 더 예뻐!
[구매하기 >](#)

AD 네이버 리빙원도

[쇼핑](#) [상품](#) [쇼핑몰](#) [MEN](#)

G마켓 · 옥션 · 11번가 · 위메프 · 티몬 · 이마트몰 · 신세계몰
GS샵 · 롯데몰 · 롯데닷컴 · CJmall · 현대H몰 · 올리브영



LIKE LION

HTML로

정보 보여주기는 가능
하지만 그 정보는 어떻게?



LIKE LION

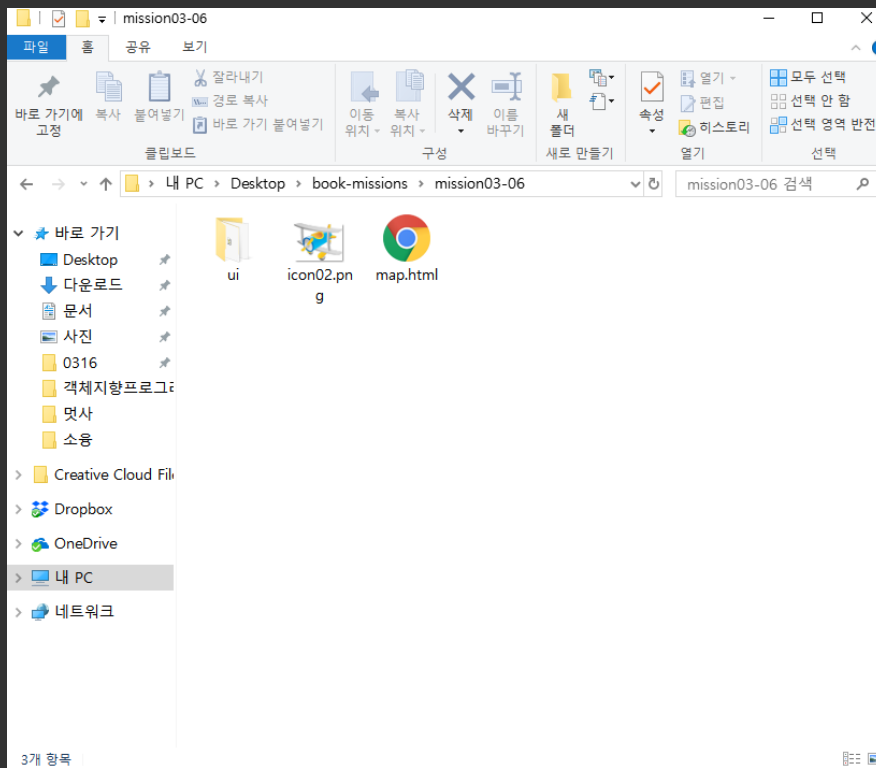
HTML / CSS로 만든 웹 페이지

정적이다



LIKE LION

지금까지의 한계



내 PC로
다운로드



LIKE LION

지금까지의 **한계**



입력은 어떻게?
그리고 저장은?

The image shows the Naver login interface. At the top is the blue Naver logo. Below it, on the right, is the text '단체아이디 로그인'. There are two input fields: the first is labeled '아이디' (ID) and the second is labeled '비밀번호' (Password). Below these fields is a large blue button with the white text '로그인' (Login). At the bottom left, there is a checked checkbox labeled '로그인 상태 유지' (Keep login state). At the bottom right, there is a label 'IP보안 ON' and a link '일회용 로그인 ?'.

LIKE LION

지금까지의 **한계**



방대한 양의
데이터를
html에 담기?



LIKE LION

멋사가 만들 웹 서비스



LIKE LION

멋사가 만들 웹 서비스

Web Service

World Wide Web (=인터넷)



LIKE LION

멋사가 만들 웹 서비스

Web Service

사용자가 원하는 것을 제공



LIKE LION

멋사가 만들 웹 서비스

Web Service

인터넷으로 사용자가 원하는 것을
제공하는 것



LIKE LION

멋사가 만들 웹 서비스



Web Service

인터넷으로 사용자가 원하는 것을
제공하는 것



LIKE LION

Web Service



검색결과



사진



동영상



LIKE LION

요청

응답



LIKE LION



요청



응답



LIKE LION



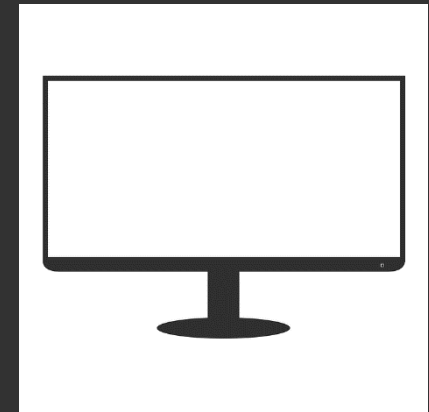
URL (Uniform Resource Locator)



요청



응답



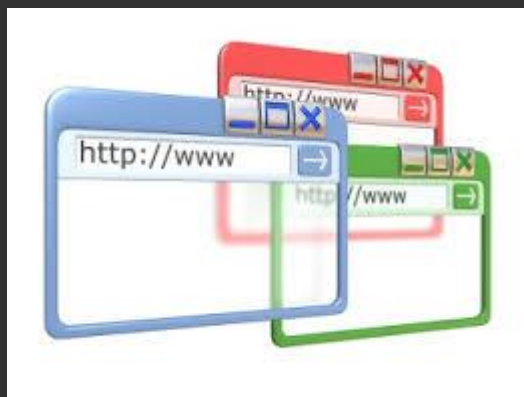
LIKE LION



요청



응답



HTML

요청에 대한 정보가 들어간 html파일



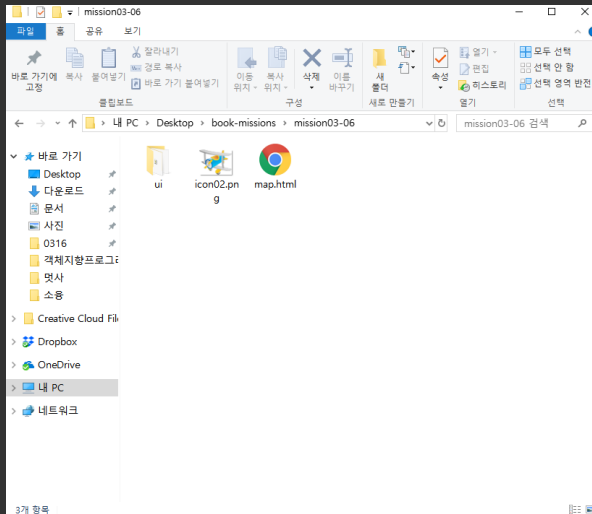
LIKE LION

HTML + Ruby on rails



LIKE LION

솔루션!



웹 브라우저로 보고싶어!



LIKE LION

솔루션!

```
class Post < ActiveRecord::Base
  has_many :comments
end

class Comment < ActiveRecord::Base
  def self.search(query)
    find(:all, :conditions => [ "body = ?", query ])
  end
end

# SELECT * from comments WHERE post_id = 1 AND body = 'hi'
Post.find(1).comments.search "hi"
```

입력 받은 정보를

루비 문법으로

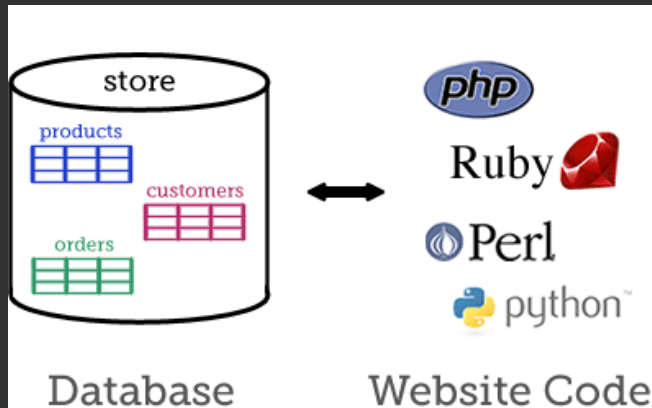
원하는 대로 저장

```
!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Str
head>
<meta http-equiv="Content-Type" content="text/h
<title>CSS3</title>
<link href="style.css" rel="stylesheet" type="c
</head>
<body>
<div id="container"> <!-- Main Container -->
<div class="menu"> <!-- Menu here --></div>
<div class="article"><h2>CSS3 Sample</h2></div>
<div class="4">CSS3 & HTML5 are so good!</a>
```



LIKE LION

솔루션!



데이터베이스(DB)
원하는 정보만
가져오기!



LIKE LION

멋사가 만들 웹 서비스

사용자와 요청과 응답

다수의 사용자와 소통

다양한 데이터와 연산



LIKE LION



루비 VS 루비온 레일즈



LIKE LION

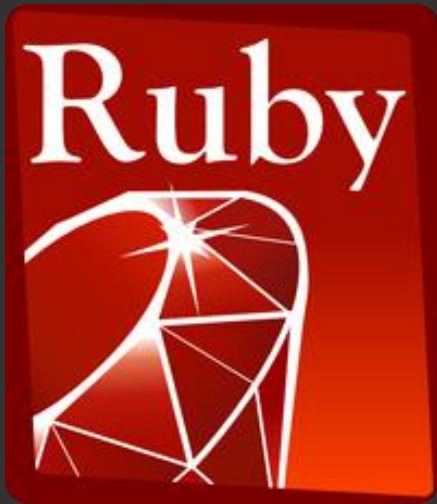


루비 (Ruby)

프로그래밍 언어



LIKE LION



루비 (Ruby)

프로그래밍 언어

반복, 조건, 함수, 메소드 등

유저에게 받아온 정보를 가공



LIKE LION

루비온 레일즈 (Ruby on Rails)

프레임워크



웹 개발을 좀더 편하게 해준다.

웹 개발의 기본적인 구조, 코드 제공



LIKE LION

NOT



웹서버 구축,
DB연결, Ruby와 Html의 연결
모두 내가 직접 코딩!!



LIKE LION



초보자가 배우기 쉽다.

인기있는 풀 스택 프레임 워크

가장 빠르게 내가 원하는 걸 만들 수 있다.



LIKE LION

Part 3 . READY



LIKE LION

개발을 위한 환경 준비



AWS Cloud9은 브라우저만으로 코드를 작성, 실행 및 디버깅할 수 있게 해주는 클라우드 기반 통합 개발 환경(IDE)이며, 코드 편집기, 디버거 및 터미널이 포함되어 있습니다. Cloud9은 JavaScript, Python, PHP를 비롯하여 널리 사용되는 프로그래밍 언어를 위한 필수 도구가 사전에 패키징되어 제공되므로, 새로운 프로젝트를 시작하기 위해 파일을 설치하거나 개발 머신을 구성할 필요가 없습니다. Cloud9 IDE는 클라우드 기반이므로, 인터넷이 연결된 머신을 사용하여 사무실, 집 또는 어디서든 프로젝트 작업을 할 수 있습니다. 또한, Cloud9은 서버리스 애플리케이션을 개발할 수 있는 원활한 환경을 제공하므로 손쉽게 서버리스 애플리케이션의 리소스를 정의하고, 디버깅하고, 로컬 실행과 원격 실행 간에 전환할 수 있습니다. Cloud9에서는 개발 환경을 팀과 신속하게 공유할 수 있으므로 프로그램을 연결하고 서로의 입력값을 실시간으로 추적할 수 있습니다.



LIKE LION



Local로 했을 때

각종 프로그램 설치

개발 환경 완성

바로 개발 결과물 공유 가능

인터넷 연결이 느슨하면 개발 힘들



LIKE LION

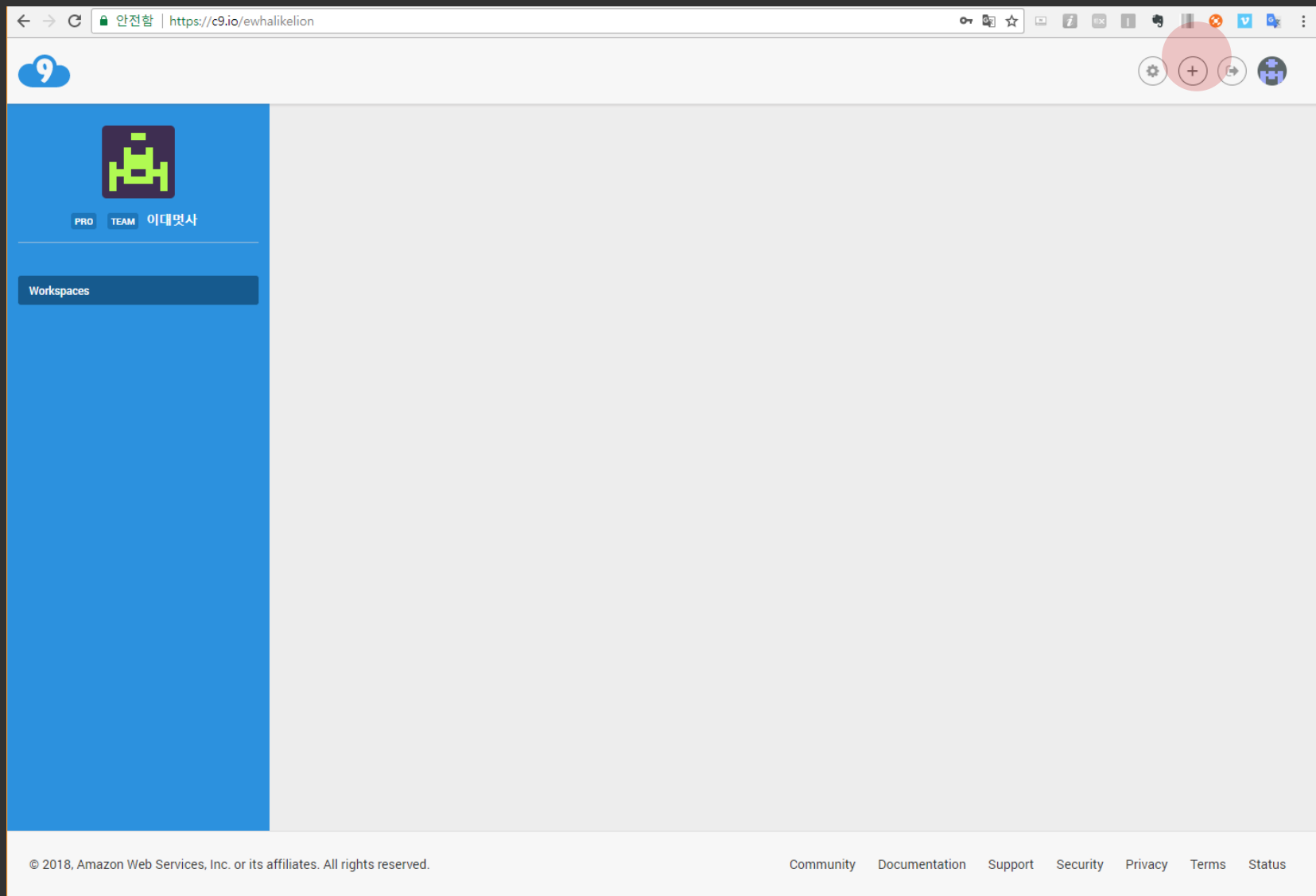
각자 메일에서 C9 invite 링크 클릭



링크 즐겨찾기 부탁드립니다



LIKE LION



Create a new workspace

| Workspace name | Description |
|----------------|-------------|
| minjeong_test | 0327 세션 테스트 |

Team

[Hosted workspace](#) [Clone workspace](#) [Remote SSH workspace](#) [Salesforce](#)

Private

This is a workspace for your eyes only

Public

This will create a workspace for everybody to see

Clone from Git or Mercurial URL (optional)

e.g. `ajaxorg/ace` or `git@github.com:ajaxorg/ace.git`

Choose a template

The screenshot shows the Heroku 'Choose a template' page. At the top, it says 'Choose a template'. Below this is a grid of 10 application templates. The 'Ruby' template, which features the Rails logo, is highlighted with a red border and a light red background. Other templates include HTML5, Node.js, PHP, Python, Django, C++, Wordpress, Rails Tutorial, Blank, and Harvard's CS50.

[Create workspace](#)



Cloud9 IDE interface showing a Rails project setup. The browser address bar displays the URL: https://ide.c9.io/mjung1798/minjeong_test#openfile-README.md.

The interface includes a menu bar (File, Edit, Find, View, Goto, Run, Tools, Window, Support, Preview, Run Project), a workspace sidebar with a file explorer, and a main editor area displaying the README.md file.

Annotations highlight key features:

- Folder**: Points to the file explorer sidebar.
- Run Project (URL 생성)**: Points to the "Run Project" button in the menu bar.
- Bash (Console)**: Points to the terminal window at the bottom.

The README.md content includes:

README

Welcome to your Rails project on Cloud9 IDE!

To get started, just do the following:

1. Run the project with the "Run Project" button in the menu bar on top of the IDE.
2. Preview your new app by clicking on the URL that appears in the Run panel below (<https://HOSTNAME/>).

Happy coding! The Cloud9 IDE team

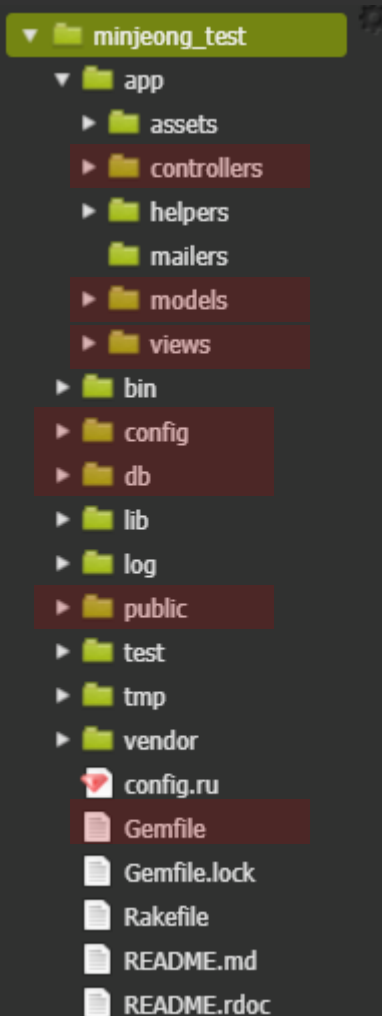
Support & Documentation

Visit <http://docs.c9.io> for support, or to learn more about using Cloud9 IDE visit <http://www.youtube.com/user/c9ide>

The terminal window shows the prompt: `mjung1798:~/workspace $`



LIKE LION



다음세션 예고

C9 에서 Rails 실습!

◀ 자주 사용할 폴더 & 파일



LIKE LION