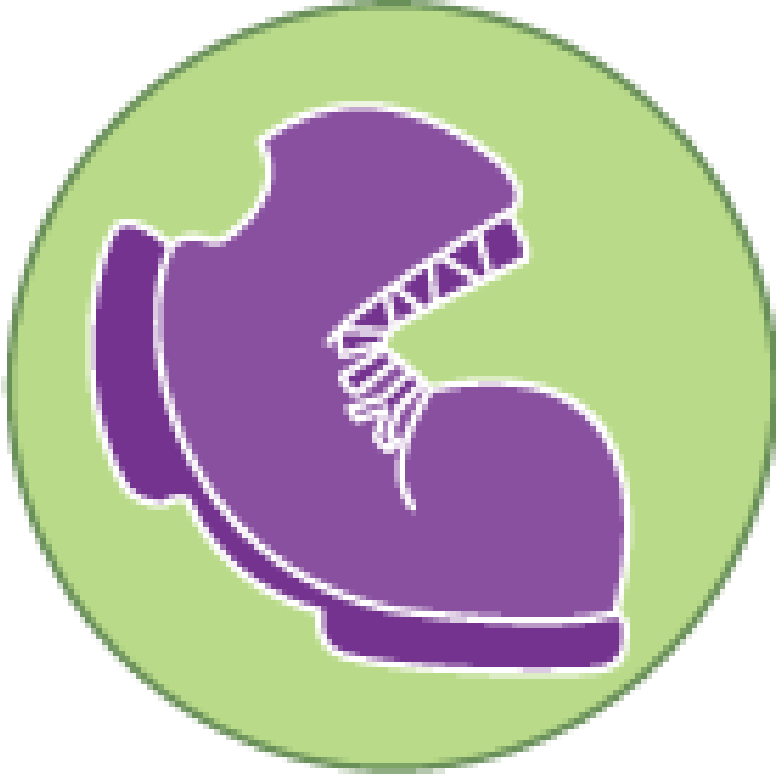


Name_____



ÖNYÜKLEME 2

DERİS 1

	Racket Kodu	Pyret Kodu
Numbers	(define YAŞ 14) (define BİR-SAYI 0.6) (define HİZ -90)	YAŞ = 14 BİR-SAYI = 0.6 HİZ = -90 İki tane de kendiniz yazın:
Metinler	(define SINIF "Önyükleme") (define CÜMLE "Kodlamak eğlencelidir!") (define BİR_METİN "2500")	SINIF = "Önyükleme" CÜMLE = "Kodlamak eğlencelidir!" BİR_METİN = "2500" İki tane de kendiniz yazın:

	Racket Kodu	Pyret Kodu
	<pre>(define ŞEKİL (triangle 40 "outline" "red")) (define ÇERÇEVE (star 80 "solid" "green")) (define KARE (rectangle 50 50 "solid" "blue"))</pre>	<pre>ŞEKİL = triangle(40, "outline", "red") ÇERÇEVE = star(80, "solid", "green") KARE = rectangle(50, 50, "solid", "blue") One of your own:</pre>
Mantıksal	<pre>(define MANTIKSAL true) (define MANTIKSAL 2 false)</pre>	<pre>MANTIKSAL = true One of your own:</pre>
Fonksiyonlar	<pre>; ikikatı : Sayı -> Sayı ; Verilen sayıyı iki ile çarp (EXAMPLE (ikikatı 5) (* 2 5)) (EXAMPLE (ikikatı 7) (* 2 7)) (define (ikikatı n) (* 2 n))</pre>	<pre>#ikikatı : Sayı -> Sayı # Verilen sayıyı iki ile çarp examples: ikikatı(5) is 2 * 5 ikikatı(7) is 2 * 7 end fun ikikatı(n): 2 * n end</pre>

Hızlı Fonksiyonlar!

Aşağıda fonksiyonlar için sözleşmeyi doldurun, sonra iki tane örnek ve tanımı da kendiniz yapmaya çalışın.

ikikatı: Sayı → Sayı

 isim girdi çıktı alanı

examples:

ikikatı (5) is 2 * 5

ikikatı (7) is 2 * 7

end

fun ikikatı (n):

2 * n

end

#

:

→

 isim girdi çıktı alanı

examples:

() is

() is

end

fun _____ (_____):

end

Hızlı Fonksiyonlar!

Her fonksiyon için sözleşmeyi doldurun, daha sonra iki örnek ve fonksiyon tanımını yazmaya çalışın.

#	:	→
_____	_____	_____
isim	girdi	çıktı alanı

examples:

_____	(_____)	is
_____	(_____)	is
_____		_____		_____

end

fun _____ (_____):

end

#	:	→
_____	_____	_____
isim	girdi	çıktı alanı

examples:

_____	(_____)	is
_____	(_____)	is
_____		_____		_____

end

fun _____ (_____):

end

Hızlı Fonksiyonlar!

Her fonksiyon için sözleşmeyi doldurun, daha sonra iki örnek ve fonksiyon tanımını yazmaya çalışın.

#	:	→
_____	_____	_____
isim	girdi	çıktı alanı

examples:

_____	(_____)	is
_____	(_____)	is
_____	_____	_____	_____	_____

end

fun _____ (_____):

end

#	:	→
_____	_____	_____
isim	girdi	çıktı alanı

examples:

_____	(_____)	is
_____	(_____)	is
_____	_____	_____	_____	_____

end

fun _____ (_____):

end

Bug Hunting: Pyret Edition

#1	<pre>SECONDS = (7) STRING = my string</pre>	
#2	<pre>SHAPE1 = circle(50 "solid" "blue") SHAPE2 = triangle(75, outline, yellow)</pre>	
#3	<pre># triple : Number -> Number # Multiply a given number by # 3 to triple it examples: triple(5) = 3 * 5 triple(7) = 3 * 7 end</pre>	
#4	<pre>fun triple(n): 3 * n</pre>	
#5	<pre># ys : Number -> Number # Given a number, create a solid # yellow star of the given size examples: ys(99) is star(99, "solid", "yellow") ys(33) is star(99, "solid", "yellow") ys(size): star(size "solid" "yellow") end</pre>	

DER S 2

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Problem: iki-kat-yarıçap

iki-kat-yarıçap isimli bir fonksiyon yazın, öyle ki bu fonksiyon girdi olarak bir yarıçap ve renk alsın. Sonra istenen renkte ve verilenin iki katı yarıçapa sahip çerçeve bir çember üretsın.

Sözleşme ve Amaç İfadesi

Sözleşmenin üç kısmı bulunur:

```
#           :           →  
_____  
isim       girdi       çıktı alanı  
  
# _____  
fonksiyon ne yapıyor?
```

Örnekler

Fonksiyonun nasıl çalışacağına dair örnekler üretin.

examples:

```
           (           ) is  
_____  
           (           ) is  
_____  
_____
```

end

Fonksiyon tanımı

Örnekler arasında değişenin ne olduğunu belirledikten sonra değişmeyen herşeyi kopyalayın, değişenlerin yerine ise seçtiğiniz değişken isimlerini koyun.

fun _____ (_____):

end

Problem: iki-kat-genişlik

iki-kat-genişlik isimli bir fonksiyon yazın, öyle ki bu fonksiyon girdi olarak bir sayı alsın (dikdörtgenin yüksekliği). Sonra yüksekliği o sayı ve genişliği onun iki katı olan bir dikdörtgen üretsinsin.

Sözleşme ve Amaç İfadesi

Sözleşmenin üç kısmı bulunur:

```
#           :           →  
_____  
isim       girdi       çıktı alanı  
  
# _____  
fonksiyon ne yapıyor?
```

Örnekler

Fonksiyonun nasıl çalışacağına dair örnekler üretin.

examples:

```
           (           ) is  
_____  
           (           ) is  
_____  
_____
```

end

Fonksiyon tanımı

Örnekler arasında değişenin ne olduğunu belirledikten sonra değişmeyen herşeyi kopyalayın, değişenlerin yerine ise seçtiğiniz değişken isimlerini koyun.

fun _____ (_____):

end

Problem: next-position

next-position isimli bir fonksiyon yazın. Paraşütçünün x ve y koordinatlarını alsın, x koordinatını 5 arttırıp y koordinatını da 5 eksilterek bir Coord oluşturup geri döndürsün..

Contract+Purpose Statement

Sözleşmenin üç kısmı bulunur:

```
# _____ : _____ →  
  isim          girdi          çıktı alanı  
  
# _____  
fonksiyon ne yapıyor?
```

Örnekler

Fonksiyonun nasıl çalışacağına dair örnekler üretin.

examples:

```
_____ ( _____ ) is  
_____ ( _____ ) is  
_____ ( _____ )
```

end

Fonksiyon tanımı

Örnekler arasında değişenin ne olduğunu belirledikten sonra değişmeyen herşeyi kopyalayın, değişenlerin yerine ise seçtiğiniz değişken isimlerini koyun.

fun _____ (_____):

```
  _____  
end
```

Veri Yapısı

Bir kek çeşit/tat, renk, mesaj, katmanlar, ve dondurmali-olup-olmadigi bilgisinden oluşur

data Cake:

```
|    cake(_____
```

```
_____
```

```
_____
```

```
_____
```

```
_____)
```

end

Bu yapının örneklerini oluşturmak için şu kodu yazalım:

```
kek1 = _____
```

```
kek2 = _____
```

kek2'nin alanlarına ulaşmak için şu kodu yazarım:

```
_____
```

```
_____
```

```
_____
```

```
_____
```

```
_____
```

DER S 3

[illegible]

Veri Yapısı

Bir Parti yer/location, tema/theme, ve misafir-sayısı/number-of-guests bilgisinden oluşur

data Party:

```
| party(_____  
_____  
_____  
_____  
_____)
```

end

Bu yapının örneklerini oluşturmak için şu kodu yazalım:

```
party1 = _____  
party2 = _____
```

party2'nin alanlarına ulaşmak için şu kodu yazarım:

```
_____  
_____  
_____  
_____  
_____
```

Problem: change-flavor

change-flavor isimli bir fonksiyon yazın. Bir Cake ve bir metin alsın ve verilenle aynı ancak sadece çeşidi istenilene uygun değiştirilmiş bir Cake döndürsün.

Sözleşme ve Amaç İfadesi

```
# _____ : _____ → _____  
# _____
```

Örnekler

examples:

```
_____ ( _____ ) is  
_____  
_____  
_____  
_____ ( _____ ) is  
_____  
_____  
_____  
_____
```

end

Fonksiyon tanımı

Örnekler arasında değişenin ne olduğunu belirledikten sonra değişmeyen herşeyi kopyalayın, değişenlerin yerine ise seçtiğiniz değişken isimlerini koyun.

Fun _____ (_____):

```
_____  
_____  
_____
```

end

Problem: erir-mi

erir-mi isimli bir fonksiyon yazın. Bir Cake ve bir sıcaklık değeri alsın ve hem sıcaklık 32 dereceden yüksek hem de Cake dondurmalı ise 'true/doğru' değerini döndürsün, verilemeyenle aynı ancak sadece çeşidi istenilene uygun değiştirilmiş bir Cake döndürsün.

Sözleşme ve Amaç İfadesi

```
# _____ : _____ → _____  
# _____
```

Örnekler

Fonksiyonun nasıl çalışacağına dair örnekler üretin.

examples:

```
_____ ( _____ ) is  
_____  
-----  
-----  
-----  
_____ ( _____ ) is  
_____  
-----  
-----  
-----
```

end

Fonksiyon tanımı

Örnekler arasında değişenin ne olduğunu belirledikten sonra değişmeyen herşeyi kopyalayın, değişenlerin yerine ise seçtiğiniz değişken isimlerini koyun.

Fun _____ (_____):

```
_____  
_____  
_____
```

end

DERS 4

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

DER S 5

[illegible]

Problem: keypress (Ninja Dünyası)

Ninja dünyasındaki her tuşa basma için (keypress <dünya> <tuş>) dünyayı nasıl değiştirmeli gösterin

Sözleşme ve Amaç İfadesi

```
# keypress: World String → World
# Given a world and a key, produce a new world with
NinjaCat's position
# moved by 10 pixels, depending on which arrow key was
pressed
```

Örnekler

examples:

```
    keypress(worldA, "up") is
        world(worldA.dogX, worldA.coinX, worldA.catX, worldA.catY + 10)
    keypress(worldB, "down") is
        world(worldB.dogX, worldB.coinX, worldB.catX, worldB.catY - 10)
    keypress(worldA, "left") is
        world(worldA.dogX, worldA.coinX, worldA.catX - 10, worldA.catY)
    keypress(worldB, "right") is
        world(worldB.dogX, worldB.coinX, worldB.catX + 10, worldB.catY)
end
```

Fonksiyon tanımı

```
fun keypress(current-world, key) :
  ask:
    | string-equal(key, "up") then:
      world(current-world.dogX, current-world.coinX,
            current-world.catX, current-world.catY + 10)
    | string-equal(key, "down") then:
      world(current-world.dogX, current-world.coinX,
            current-world.catX, current-world.catY - 10)
    | string-equal(key, "left") then:
      world(current-world.dogX, current-world.coinX,
            current-world.catX - 10, current-world.catY)
    | string-equal(key, "right") then:
      world(current-world.dogX, current-world.coinX,
            current-world.catX + 10, current-world.catY)
    | otherwise: current-world
  end
end
```


Problem: next-world (Ninja Dünyası)

Verilen bir dünya için dogX'e 10 ekleyen, coinX'ten 5 çıkartan, ve sadece kedinin y koordinatı 75ten büyükse o koordinattan 5 çıkartan bir next-world fonksiyonu yapın.

Sözleşme ve Amaç İfadesi

```
# _____ : _____ → _____  
# _____
```

Örnekler

Fonksiyonun nasıl çalışacağına dair örnekler üretin.

examples:

```
_____ ( _____ ) is _____  
-----  
_____ ( _____ ) is _____  
-----
```

end

Fonksiyon tanımı

Örnekler arasında değişenin ne olduğunu belirledikten sonra değişmeyen herşeyi kopyalayın, değişenlerin yerine ise seçtiğiniz değişken isimlerini koyun.

```
fun _____ ( _____ ) :  
  ask: _____ then:  
    _____  
  | otherwise:  
    _____  
end  
end
```

DERS 6

[illegible]

Problem: red-shape

Bir şeklin ismini alan ("circle", "triangle", "star", "rectangle" gibi) ve o şekli çizen red-shape isimli bir fonksiyon yazın. Bütün şekiller kırmızı ve içi dolu olmalı, ve istediğiniz boyda olabilirler.

Sözleşme ve Amaç İfadesi

```
# _____ : _____ → _____  
# _____
```

Örnekler

Fonksiyonun nasıl çalışacağına dair örnekler üretin.

examples:

```
_____ ( _____ ) is _____  
_____ ( _____ ) is _____  
_____ ( _____ ) is _____  
_____ ( _____ ) is _____  
end
```

Fonksiyon tanımı

```
fun _____ ( _____ ) :  
  ask:  
    | _____ then:  
    _____  
    | _____ then:  
    _____  
    | _____ then:  
    _____  
  end  
end
```

Problem: strong-password

Websiteleri için güvenli şifre kullanmak gerekir. Bir kullanıcı adı ve bir şifre alan, ve ikisinin aynı olmadığını kontrol edip sonra da password'ün uzunluğunun (string-length) 8'den büyük olması için kontrol eden strong-password isimli bir fonksiyon yazın. Fonksiyon kul şifresinin yeteri kadar güçlü olup olmadığını bildiren bir mesaj döndürmeli.

Sözleşme ve Amaç İfadesi

```
# _____ : _____ → _____  
# _____
```

Örnekler

Fonksiyonun nasıl çalışacağına dair örnekler üretin.

examples:

```
_____ ( _____ ) is _____  
_____ ( _____ ) is _____  
_____ ( _____ ) is _____
```

end

Fonksiyon tanımı

```
fun _____ ( _____ ) :  
  ask: _____ then:  
    _____  
  | _____ then:  
    _____  
  | otherwise: _____  
end  
end
```


Yardımcı Fonksiyonları İKurmak

is-off-right : _____ → _____

examples:

_____ (_____) is _____

_____ (_____) is _____

end

fun _____ (_____):

end

is-off-left : _____ → _____

examples:

_____ (_____) is _____

_____ (_____) is _____

end

fun _____ (_____):

end

_____ : _____ → _____

examples:

_____ (_____) is _____

_____ (_____) is _____

end

fun _____ (_____):

end

_____ : _____ → _____

examples:

_____ (_____) is _____

_____ (_____) is _____

end

fun _____ (_____):

end

DERS 7

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Problem: line-length

İki sayı alan ve ikisinin farkını döndüren line-length isimli bir fonksiyon yazın. Fonksiyon her zaman küçük sayıyı büyük olandan çıkartmalı.

Sözleşme ve Amaç İfadesi

```
# _____ : _____ → _____  
# _____
```

Örnekler

examples:

```
_____ ( _____ ) is  
_____
```

```
-----  
_____ ( _____ ) is  
_____
```

end

Fonksiyon tanımı

```
fun _____( _____ ) :  
  _____:
```

```
  _____  
  _____  
  _____  
  _____  
  _____
```

end

end

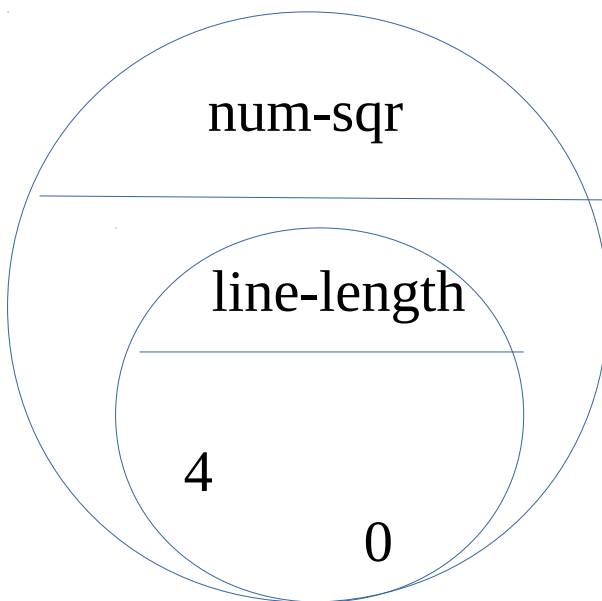
Uzaklık

Oyuncu (4,2) noktasında ve hedef te (0,5) noktasında. Uzaklık fonksiyonu oyuncunun x ve y, ve hedef karakterin x ve y koordinatlarını girdi olarak alacak.

Örneği doldurmak için aşağıdaki formülü kullanın:

$$\sqrt{(line-length\ 4\ 0)^2 + (line-length\ 2\ 5)^2}$$

Bunu bir değerlendirme çemberine dönüştürün (bir kısmını sizin için yaptık



Onu Pyret koduna dönüştürün:

Problem: distance/uzaklık

distance isimli bir fonksiyon yazın ve dört girdi alsın:

- ox: oyuncunun x koordinatı
- oy: oyuncunun y koordinatı
- kx: başka bir oyun karakterinin x koordinatı
- ky: karakterin y koordinatı

Fonksiyon iki nokta arasındaki uzaklığı aşağıdaki formülü kullanarak döndürmelidir:

$$Distance = \sqrt{(line-length\ ox - kx)^2 + (line-length\ oy - ky)^2}$$

Sözleşme ve Amaç İfadesi

_____ : _____ → _____

Örnekler

examples:

(_____) is

(_____) is

end

Fonksiyon tanımı

fun _____(_____) :

end

end

Problem: is-collision / çarpışma-mı

is-collision isimli bir fonksiyon yazın ve dört girdi alsın:

- ox: oyuncunun x koordinatı
- oy: oyuncunun y koordinatı
- kx: başka bir oyun karakterinin x koordinatı
- ky: karakterin y koordinatı

Fonksiyon iki nokta arasındaki uzaklığı 50 pikselden az ise true döndürmeli, diğer durumlarda false döndürmelidir.

Sözleşme ve Amaç İfadesi

_____ : _____ → _____

Örnekler

examples:

(_____) is

(_____) is

end

Fonksiyon tanımı

fun _____(_____) :

end

end