



Modul I. Lesson 9

String I

Nasibali Abdiyev. Flutter Development

Repeat the previous lesson

- Function
- Function type
- Parameter
- Argument
- Return
- Return type

Plan

- String declaration
- String is constant
- Unicode
- CodeUnit
- Searching inside a string
- Some methods of String

String



 main.dart

```
void main() {  
  String singleQuote = 'This text is enclosed in single quotation marks.';  
  String doubleQuotes = "This text is enclosed in double quotation marks.";  
  String multiLineText = """  
This text is displayed as written  
""";  
  
  String parts = "Part1." "Part2."  
    "Part3";  
  
  String tab = "Start\tEnd";  
  String endlime = "Start\nEnd";  
  
  String symbol$ = "\$ 100";  
  
  String symbolSingleQuote1 = '0\'zbekiston';  
  String symbolSingleQuote2 = "0'zbekiston";  
  
  String symbolDoubleQuotes1 = "symbol: (\")";  
  String symbolDoubleQuotes2 = 'symbol: (")';  
  
  String unformattedText = r'Script for String: "$" "\n" "\t".';  
}
```

As a String list



main.dart

```
void main() {  
    String text = "PDP Academy";  
    for(int i = 0; i < text.length; i++) {  
        print("${i + 1} - ${text[i]}");  
    }  
}
```

Add strings



main.dart

```
void main() {  
    String text1 = "Hello";  
    String space = " ";  
    String text2 = "World";  
    String exclamation = "!";  
  
    String result = text1 + space + text2 + exclamation;  
    print(result);  
}
```

Multiplication of strings

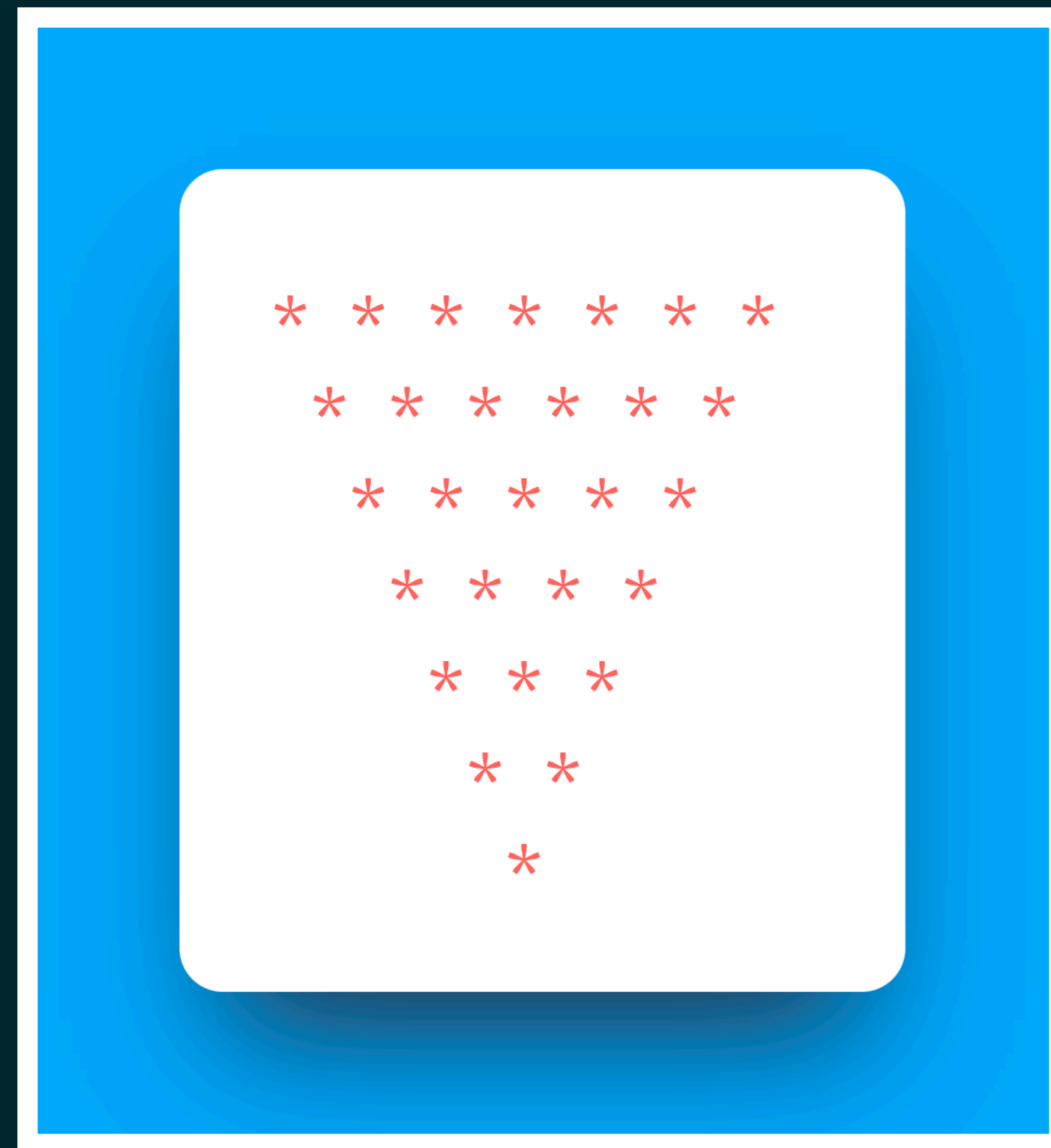


main.dart

```
void main() {  
  String starWithSpace = '* '  
  String space = " "  
  int column = 7;  
  
  for(int i = 0; i < column; i++) {  
    String row = space * (column - i) + starWithSpace * i;  
    print(row);  
  }  
}
```


Exercise

- Yuqorida ko'rsatilgan misoldan foydalanib quyidagi shaklni console ga chiqaruvchi dastur tuzing.



Unicode

- Unicode – bu butun dunyo bo'ylab har xil yozuv tizimlaridagi har bir belgi uchun noyob raqamli identifikator ajratib beruvchi standartdir. Bu standart turli dasturlash tillari va operatsion tizimlarida bir xil tarzda har qanday matnli ma'lumotlarni ifodalash va almashish imkonini beradi.
- Unicode, global miqyosda matnli ma'lumotlarni ishlov berish va turli tillarda ish yuritish imkonini beruvchi universal yechim sifatida qabul qilingan. Bu standart, xalqaro miqyosda raqamli kommunikatsiya va ma'lumotlar almashinuvini soddalashtirishda muhim rol o'ynaydi.
- Dart satri (String ob'ekti) UTF-16 kod birliklari ketma-ketligini o'z ichiga oladi.
- UTF-16, "Unicode Transformation Format - 16-bit" deb nomlangan, Unicode belgilarini kodlashning bir usulidir. Bu, Unicode belgilarini ifodalash uchun 16-bit (yoki ikkita bayt) ishlatadi.

Features

- **Global Qamrov:** Unicode deyarli barcha mavjud yozuv tizimlarini, shu jumladan lotin, kirill, arab, xitoy, yapon, koreys va boshqa ko'plab tillarning belgilarini qamrab oladi.
- **Noyob Identifikatorlar:** Har bir belgi o'ziga xos raqamli kodga (kodek point) ega bo'lib, bu raqam butun dunyo bo'ylab bir xil.
- **Keng Qo'llanilishi:** Unicode hozirgi zamonaviy dasturlash tillari, ma'lumotlar bazalari va fayl formatlari kabi ko'plab texnologiyalarda qo'llaniladi.
- **Bir-biriga Moslashuvchanlik:** Unicode, eski kodlash tizimlari bilan moslashuvchanlikni ta'minlaydi, masalan, ASCII kodlar to'plami Unicode tizimida to'g'ridan-to'g'ri muvofiqlashtirilgan.
- **Kengaytirilishi:** Unicode doimiy ravishda yangi belgilar va yozuv tizimlari bilan yangilanib turadi, bu esa uni zamonaviy va kelajakdagi talablarga moslashuvchan qiladi.

codeUnitAt(int index) → int

- Berilgan indeksdagi 16-bitli UTF-16 kod birligini qaytaradi.



main.dart

```
void main() {  
  const text = "Flutter 🎯";  
  
  for (int i = 0; i < text.length; i++) {  
    print("${text[i]} = ${text.codeUnitAt(i)}");  
  }  
}
```

Exercise

1. Ism familiyangizni unicodelarini toping. `codeUnitAt` methodi orqali qilish tavsiya etiladi.
2. Kodlar jadvalida kiritilgan belgidan oldin va keyin turuvchi belgilarni chigaruvchi dastur tuzilsin.

codeUnits → List<int>

- Ushbu satrning UTF-16 kod birliklarining o'zgartirilmaydigan ro'yxati.

```
void main() {  
  const text = "Flutter 🎯";  
  // "🎯" - [55356, 57263]  
  
  final list = text.codeUnits;  
  
  for (int symbol in list) {  
    print(symbol);  
  }  
}
```

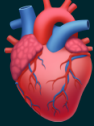

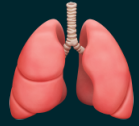

Convert to each other



main.dart

```
void main() {  
  String text = "hello";  
  
  List<int> unicodes = text.codeUnits;  
  print(unicodes);  
  
  String result = String.fromCharCode(unicodes);  
  print(result);  
}
```


Exercise

- 97,98,99...121,122 unicodelar berilgan bo'lsa bu ketma ketlikni aslini toping.
- Quyidagi textning faqat lotin tilidagi harflarining Unicodeni toping.
 - String text = "Heart - , Brain - , Lungs - , Leaf - ".

Unicode characters in the range 32-127

- katta harf 65-90 oralig'ida
- kichik harf 97-122 oralig'ida
- raqam 48-57 oralig'ida
- maxsus belgilar boshqa diapazonlarda

32		48	0	64	@	80	P	96	`	112	p
33	!	49	1	65	A	81	Q	97	a	113	q
34	"	50	2	66	B	82	R	98	b	114	r
35	#	51	3	67	C	83	S	99	c	115	s
36	\$	52	4	68	D	84	T	100	d	116	t
37	%	53	5	69	E	85	U	101	e	117	u
38	&	54	6	70	F	86	V	102	f	118	v
39	'	55	7	71	G	87	W	103	g	119	w
40	(56	8	72	H	88	X	104	h	120	x
41)	57	9	73	I	89	Y	105	i	121	y
42	*	58	:	74	J	90	Z	106	j	122	z
43	+	59	;	75	K	91	[107	k	123	{
44	,	60	<	76	L	92	\	108	l	124	
45	-	61	=	77	M	93]	109	m	125	}
46	.	62	>	78	N	94	^	110	n	126	~
47	/	63	?	79	O	95	_	111	o	127	DEL

Exercise

1. n butun soni berilgan ($1 \leq n \leq 26$). Lotin alifbosidagi dastlabki n ta katta harflarni chiqaruvchi dastur tuzilsin.
2. n butun soni berilgan ($1 \leq n \leq 26$). Lotin alifbosidagi n ta kichik harflarni teskari tartibda chiqaruvchi dastur tuzilsin. Ya'ni z da a gacha chiqarilsin.
3. Satr berilgan. Satring birinchi va oxirgi belgisi birlik kodini chiqaruvchi dastur tuzilsin.

Methods

Searching inside a string

- Siz satr ichida ma'lum joylarni topishingiz mumkin, shuningdek, satr ma'lum bir namuna bilan boshlanishi yoki tugashini tekshirishingiz mumkin.

```
void main() {  
    // Check whether a string contains another string.  
    print('Never odd or even'.contains('odd'));  
  
    // Does a string start with another string?  
    print('Never odd or even'.startsWith('Never'));  
  
    // Does a string end with another string?  
    print('Never odd or even'.endsWith('even'));  
  
    // Find the location of a string inside a string.  
    print('Never odd or even'.indexOf('odd') == 6);  
}
```

Exercise

1. A va B satr berilgan. A satrda B satr mavjudligini tekshiring.
2. A va B satr berilgan. A satri B satri bilan boshlanishini tekshiring.
3. Satrda berilgan symbol(belgi) necha marta qatnashganini hisoblovchi funksiya tuzing:
 - `int countSymbol(String text, String symbol) {/* your code */}`

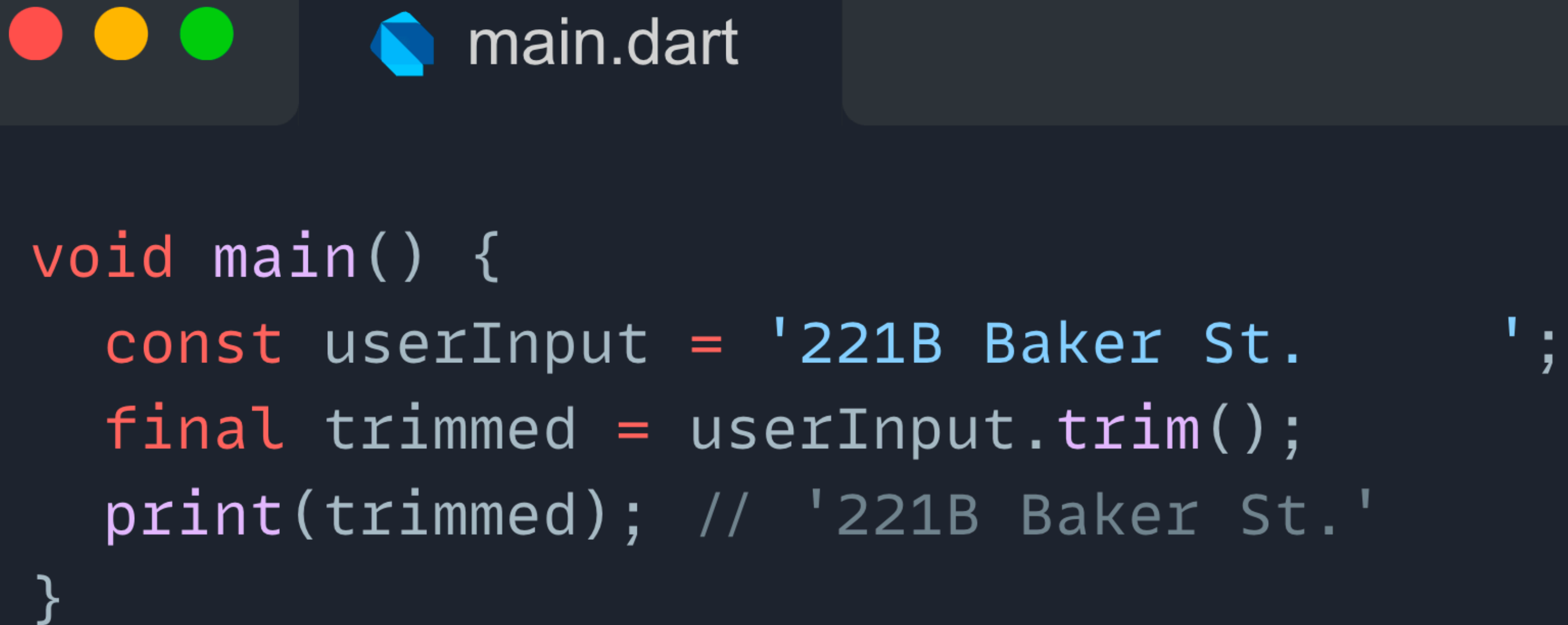
toLowerCase() → String

- Ushbu satrdagi barcha belgilarni kichik harflarga o'zgartiradi.

```
void main() {  
  String text = "My group is PDP Flutter N";  
  
  String result = text.toLowerCase();  
  print("original: $text");  
  print("update: $result");  
  
  result = text.toUpperCase();  
  print("original: $text");  
  print("update: $result");  
}
```

trim() → String

- Siz olib tashlamoqchi bo'lgan umumiy narsa - bu satr boshida yoki oxirida qo'shimcha bo'sh joy. Bo'sh joy muammoli bo'lishi mumkin, chunki ikkita satr bir xil bo'lib ko'rinishi mumkin, lekin aslida boshqacha. Ushbu bo'shliqni olib tashlash trimming deb ataladi.



```
void main() {  
    const userInput = '221B Baker St.   '  
    final trimmed = userInput.trim()  
    print(trimmed); // '221B Baker St.'  
}
```


Example



main.dart

```
void main() {  
  String text = "  My group is Flutter PDP N114  ";  
  String result = text.trim();  
  print("original: $text");  
  print("update: $result");  
  
  text = "\n\tMy group is Flutter PDP N114\t\n";  
  result = text.trim();  
  print("original: $text");  
  print("update: $result");  
}
```

Exercise

- Foydalanuvchi tomonidan kiritilgan matnni boshida va oxirida ortiqcha probellarni olib tashlovchi va matndagi hamma harflarni kichik registrlil harflarga o'tkazuvchi `sendText` nomli funksiya tuzing!

substring(int start, [int? end]) → String

- Start dan end gacha bo'lgan satrning qismini ajratib oladi.



```
void main() {  
  String str1 = "Hello World";  
  
  // from index 6 to the last index  
  print("New String: ${str1.substring(6)}");  
  // from index 2 to the 6th index  
  print("New String: ${str1.substring(2, 6)}");  
}
```

Exercise

S satr berilgan. Agar satrning uzunligi toq bo'lsa, o'rtadagi bitta belgisidan, agar juft bo'lsa, o'rtadagi 2 ta belgisidan iborat satrni ajratib oling.

Masalan:

1) `s="hovli" => javob: result="v"`

2) `s="antiga" => javob: result ="ti"`

Summary

Interview Questions

1. String ma'lumot turi qanday tur?
2. '\t' - nimani ifodalaydi?
3. '\n' - nimani ifodalaydi?
4. String ichida maxsus scriptlarni formatlamaslik uchun qaysi belgidan foydalaniladi?
5. String interpolation nima?
6. String turiga tegishli satr e'lon qilganda qatorni formatlamasligi uchun nimadan foydalanamiz? ("r");

7. String turidagi identifikatorning ixtiyoriy biror belgisiga murojaatni qaysi operator orqali amalga oshiramiz? (“[index]”)
8. String turiga tegishli identifikator bilan “+” yoki “*” operatorini qo’llasa bo’ladimi? Agar qo’llash imkoniyati bo’lsa qanday shartlar asosida foydalaniladi?
9. Unicode nima?
10. UTF-16 qanday standart?
11. String turiga tegishli codeUnitAt methodi nima vazifani bajaradi?
12. String turiga tegishli codeUnits property si nima vazifani bajaradi?
13. String.fromCharCode static methodi nima vazifani bajaradi?
14. String turiga tegishli startsWith methodi nima vazifani bajaradi?

15.String turiga tegishli endsWith methodi nima vazifani bajaradi?

16.String turiga tegishli indexOf methodi nima vazifani bajaradi?

17.String turiga tegishli trim methodi nima vazifani bajaradi?

18.String turiga tegishli subString methodi nima vazifani bajaradi?

Homework

1. Berilgan indexdagi belgini Unicode ini console ga chiqaring.

Namuna: str = "Dasturlash"

index = 0 -> D -> 68

index = 3 -> t -> 116

2. Berilgan indexdan oldingi belgini Unicode ini console ga chiqaring.

Namuna: str = "Dasturlash"

index = 1 -> D -> 68

index = 4 -> t -> 116

3. n butun soni berilgan ($32 \leq n \leq 126$). Unikodi n ga teng bo'lgan belgini consolga chiqaruvchi dastur tuzing.
4. Satr berilgan. Satrdagi katta lotin harflari sonini aniqlovchi funksiya tuzilsin.
5. Berilgan sartni katta lotin harflarini kichikga, kichiklarini katta harflarga aylantiruvchi funksiya tuzilsin. Example: “FLuTTeR” => “fIUttEr”
6. text1 va text2 satrlari berilgan. Agar text2 satri text1 satrida bo'lsa true, aks holda false chiqaruvchi dastur tuzing.
7. Str satrini tarkibida ‘A’ harfi necha marta borligini aniqlovchi dastur tuzing
8. text1 va text2 satrlari berilgan. text2 satrini text1 satrida takrorlanishlar sonini chiqaruvchi dastur tuzing.
9. Kiritilgan satrni teskari tartibda o'giruvchi return typeli funksiya tuzing tuzilsin.

10. S satr berilgan. Agar satrning uzunligi toq bo'lsa, o'rtadagi bitta belgisidan, agar juft bo'lsa, o'rtadagi 2 ta belgisidan iborat satrni ajratib oling.

Masalan:

1) `s="hovli" => javob: result="v"`

2) `s="antiga" => javob: result ="ti"`

11. str satri va c belgisi berilgan. str satrining har bir belgisidan keyin c belgisini joylashtirib yangi satr hosil qiluvchi dastur tuzing.

Masalan: `str="hello"`, `c='a'` bo'lsa, natijaviy satr `"haealalaoa"` kabi bo'ladi.

12. Butun musbat sonni ifodalovchi satr berilgan. Shu son raqamlari yig'indisini chiqaruvchi funksiya tuzilsin.

13. Satr berilgan. Satrda har 1ta uchragan '*' belgisidan 1ta oldingi va 1ta keyingi belgilar teng bo'lsa, true aks holatda false qaytaruvchi funksiya tuzing. Agar '*' dan 1ta oldin yoki 1ta keyin belgi bo'lmasa e'tiborga qarab.

sameStarChar("xy*yz") → true

sameStarChar("xy*zz") → false

sameStarChar("*xa*az") → true

14. Satr berilgan. Ushbu satrni birinchi 2ta harfini qaytaruvchi funksiya tuzing. Agar satr uzunligi 2 dan kichik bolsa, osha satrni ozini qaytaring.

firstTwo("Salom") → "Sa"

firstTwo("abcdefg") → "ab"

firstTwo("ab") → "ab"

15. Bo'sh bo'lmagan satr berilgan. Ushbu satrlarni 1- harflarini hisobga olmasdan, satrlarni qo'shib natijani qaytaruvchi funksiya tuzing.
nonStart("Hello", "There") → "ellohere", nonStart("java", "code") → "avaode", nonStart("shotl", "java") → "hotava"
16. Str satr berilgan. Shu satrdagi har "PDP" so'zidan keyin "Academy" ni qo'shib qaytaruvchi funksiya tuzing. Agar "PDP" bo'lmasa satrni o'zi qaytsin.
17. Str satr berilgan. str satrda "pdp" so'zi nechanchi indexda uchrashini aniqlab qiymat sifatida qaytaruvchi funksiya tuzing. Agar so'z topilmasa "-1" natija qaytsin.
18. N natural soni va satr berilgan. Uzunligi N ga teng bo'lgan satri shunday hosil qiling: agar kiritilgan satrning uzunligi N dan katta yoki teng bo'lsa, satrni o'zgarishsiz qoldiring; agar kiritilgan satrning uzunligi N dan kichik bo'lsa, yangi satr boshiga nuqtalarni "." Qo'shing.

19. Toq uzunlikdagi satr berilgan. Satrni o'rtasidagi 3ta harfdan iborat yangi satrni qaytaradigan funksiya tuzing. `middleThree("Candy")` → "and"
`middleThree("and")` → "and" `middleThree("solving")` → "lvi"
20. Butun sondan iborat kiritilgan `str1` va `str2` satrlarni arifmetik qo'shib qiymatni ham string toifasida qaytaradigan funksiya tuzing. Input: Output: `add("1", "2")` "3" `add("-10", "20")` "10"
21. Berilgan satrni palindrom so'z ekanligini tekshiruvchi funksiya tuzing. Ya'ni o'ngdan ham chapdan ham o'qilgandagi holati bir xil bo'ladigan so'z "palindrome" so'z bo'ladi. "palindrome" ga misol: "absba", "non".
22. Berilgan belgini katta yoki kichik harf ekanligini aniqlaydigan funksiya tuzing. Agar belgi harf bo'lmasa "harf emas", katta harf bo'lsa "katta harf", kichik harf bo'lsa "kichik harf" nomli natija qaytsin. `isLetter('H')` "Katta harf", `isLetter('h')` "Kichik harf", `isLetter('5')` "Harf emas"

Q&A

Thank you for your time!