

The left side of the slide features two large abstract geometric patterns. The top pattern is a square divided into four quadrants, each containing a different teal and white geometric design. The bottom pattern is a larger square divided into four quadrants, each containing a different teal and white geometric design, including a large circle, a semi-circle, and a triangle.

**DAY 2**

---

# **INTRO TO PROGRAMMING (PYTHON)**

---

**BY: EVAN AURELIUS**

A small abstract geometric pattern in the bottom right corner, consisting of four teal triangles arranged in a square, with a white circle in the center.

# Outline

<sup>1</sup> VARIABEL

<sup>2</sup> STRING

<sup>3</sup> STRING FORMAT

<sup>4</sup> ASCII

<sup>5</sup> BINARY

<sup>6</sup> HEXADECIMAL

<sup>7</sup> CONTROL FLOW



# PYTHON VARIABLE

- Variable digunakan untuk menyimpan data atau nilai.
- Variables dibuat dengan menetapkan nilainya.
- Nama sebuah variable itu case sensitive.

```
x = 5  
y = "Hello, World!"  
print(x)  
print(y)
```

# PYTHON DATA TYPE

Text Type:	str
Numeric Types:	int, float, complex
Sequence Types:	list, tuple, range
Mapping Type:	dict
Set Types:	set, frozenset
Boolean Type:	bool
Binary Types:	bytes, bytearray, memoryview
None Type:	NoneType

# PYTHON DATA TYPE EXAMPLE

```
# Numeric Types
x = 5          # int
y = 3.14       # float
z = 2 + 3j     # complex
```

```
# Sequence Types
a = [1, 2, 3]  # list
b = (4, 5, 6)  # tuple
c = range(0, 5) # range
```

```
# Text Type
d = "Hello, World!" # str

# Mapping Type
e = {'name': 'John', 'age': 36} # dict

# Set Types
f = {1, 2, 3} # set
g = frozenset({4, 5, 6}) # frozenset
```

```
# Boolean Type
h = True # bool

# Binary Types
i = b"Hello" # bytes
j = bytearray(5) # bytearray
k = memoryview(bytes(5)) # memoryview
```





# STRING

- Strings adalah urutan karakter.
- String diapit oleh kutip satu (") atau kutip dua (").
- Mencetak string dapat dilakukan dengan fungsi print().
- String bisa di-index, dengan karakter pertama pada index ke-0.
- Substring bisa diperoleh dengan menggunakan slicing.

```
1 str = 'Hello, World!'
2 print(str)
3 print(str[0])
4 print(str[2:5])
5 print(str[2:])
6 print(str * 2)
7 print(str + " TEST")
```

```
Hello, World!
H
llo
llo, World!
Hello, World!Hello, World!
Hello, World! TEST

=== Code Execution Successful ===
```

# STRING FORMAT

- Pemformatan string memungkinkan Anda memasukkan item ke dalam string daripada mencoba merangkai item bersama-sama menggunakan koma atau penggabungan string.
- Pemformatan dilakukan dengan menggunakan kurung kurawal {} sebagai placeholder dan memanggil metode format() pada string.
- Anda dapat menggunakan positional arguments atau keyword arguments untuk menentukan nilai yang akan menggantikan placeholder.

```
1 # Positional arguments
2 print("My name is {0}, I'm {1}".format("John",36))
3
4 # Keyword arguments
5 print("My name is {name}, I'm {age}".format(name="John",age=36))
6
7 # Combination of positional and keyword arguments
8 print("My name is {0}, I'm {age}".format("John",age=36))
```

```
My name is John, I'm 36
My name is John, I'm 36
My name is John, I'm 36
```

```
=== Code Execution Successful ===
```

# ASCII

ASCII, yang merupakan singkatan dari American Standard Code for Information Interchange, adalah standar pengkodean karakter yang digunakan untuk merepresentasikan teks dalam komputer dan perangkat komunikasi lainnya yang menggunakan teks. ASCII menggunakan angka untuk mewakili karakter tertentu. Setiap karakter diwakili oleh angka biner 7-bit yang unik, yang berarti ada 128 kemungkinan karakter (0-127).



# ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[END OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[	123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D	]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

# Binary

- Binary atau biner adalah sistem bilangan yang hanya menggunakan dua digit, yaitu 0 dan 1. Sistem ini adalah basis dari semua operasi di komputer, karena komputer bekerja dengan aliran listrik yang memiliki dua kondisi: on (1) dan off (0). Setiap angka dalam sistem biner disebut bit (binary digit). Kumpulan dari 8 bit disebut byte.
- Di dalam komputer, semua data, baik itu teks, gambar, suara, dan lain-lain, direpresentasikan dalam bentuk biner. Misalnya, huruf "A" dalam ASCII direpresentasikan sebagai 01000001 dalam biner.

2	25	
2	12	1 ← First remainder
2	6	0 ← Second Remainder
2	3	0 ← Third Remainder
2	1	1 ← Fourth Remainder
	0	1 ← Fifth Reaminder

Read Up

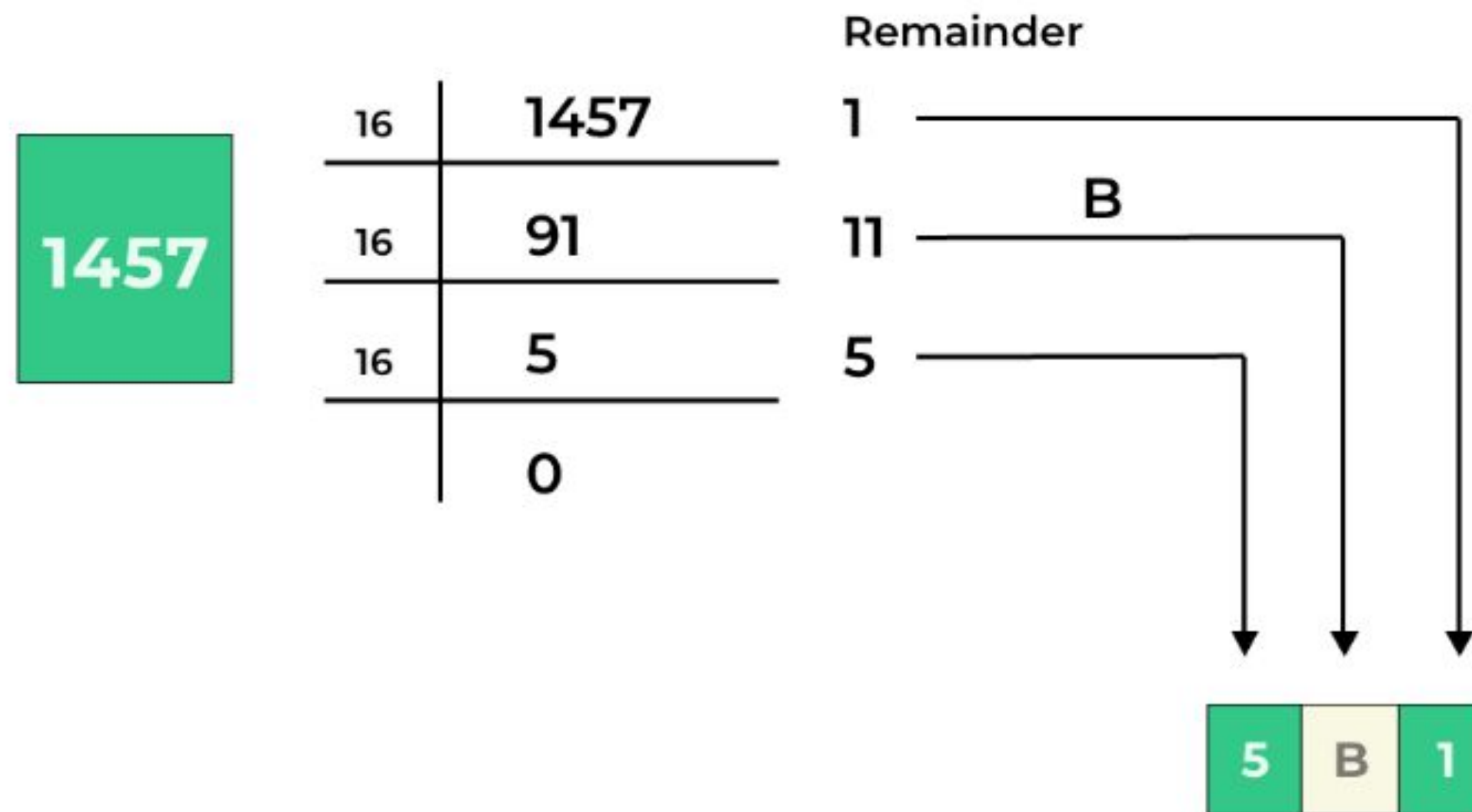
Binary Number = 11001

# Hexadecimal

Hexadecimal, atau heksadesimal, adalah sistem bilangan yang menggunakan basis 16. Sistem ini menggunakan 16 simbol untuk merepresentasikan nilai-nilai: 0-9 untuk nilai nol hingga sembilan dan A-F untuk nilai sepuluh hingga lima belas. Heksadesimal sering digunakan dalam ilmu komputer dan elektronik karena lebih ringkas dan lebih mudah dibaca oleh manusia dibandingkan dengan biner.



# Decimal to Hexadecimal



## Hexadecimal

# CONTROL FLOW –

## ARITHMETIC

- Dalam Python, kita dapat melakukan operasi-operasi aritmatika dasar seperti penjumlahan (+), pengurangan (-), perkalian (\*), pembagian (/), sisa pembagian (%), dan pemangkatan (\*\*).

Contoh:

```
a = 10
b = 5
print(a + b)  # Output: 15
print(a - b)  # Output: 5
print(a * b)  # Output: 50
print(a / b)  # Output: 2.0
print(a % b)  # Output: 0
print(a ** b) # Output: 100000
```

# CONTROL FLOW –

## COMPARISON

- Python juga mendukung operasi perbandingan seperti lebih besar (>), lebih kecil (<), lebih besar sama dengan (>=), lebih kecil sama dengan (<=), sama dengan (==), dan tidak sama dengan (!=).

Hasilnya berupa nilai Boolean (True atau False). Contoh:

```
a = 10
b = 5
print(a > b)    # Output: True
print(a < b)    # Output: False
print(a >= b)   # Output: True
print(a <= b)   # Output: False
print(a == b)   # Output: False
print(a != b)   # Output: True
```

# CONTROL FLOW – IF

- if statement digunakan untuk mengeksekusi sebuah blok kode hanya jika kondisi yang diberikan bernilai benar (True).

```
x = 10
if x > 5:
    print("x lebih besar dari 5")
```



# CONTROL FLOW – IF ELSE

- if else statement memungkinkan kita untuk menentukan blok kode alternatif yang akan dieksekusi jika kondisi pada if bernilai salah (False).

```
x = 3
if x > 5:
    print("x lebih besar dari 5")
else:
    print("x tidak lebih besar dari 5")
```

# CONTROL FLOW – IF ELIF ELSE

- if elif else statement digunakan ketika ada beberapa kondisi yang ingin diuji. elif adalah singkatan dari "else if".

```
x = 7
if x > 10:
    print("x lebih besar dari 10")
elif x > 5:
    print("x lebih besar dari 5 tetapi kurang dari atau sama dengan 10")
else:
    print("x kurang dari atau sama dengan 5")
```

---

**THANK**

**YOU**

---