

REPUBLIK INDONESIA KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

# SURAT PENCATATAN CIPTAAN

Dalam rangka pelindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan

: EC00202014254, 4 Mei 2020

Pencipta : Yuyun Yuningsih, Lusa Indah Prahartiwi, , dkk Nama Alamat Cluster Griya Adipura Sawangan Depok, Blok A3 RT. 002 RW. 004 Kel.Pengasinan Kec. Sawangan , Depok, Jawa Barat, 16518 Kewarganegaraan : Indonesia **Pemegang Hak Cipta** Nama Yuyun Yuningsih, Lusa Indah Prahartiwi, , dkk Cluster Griya Adipura Sawangan Depok, Blok A3 RT. 002 RW. 004 Alamat Kel.Pengasinan Kec. Sawangan , Depok, 8, 16518 Kewarganegaraan Indonesia Jenis Ciptaan **Program Komputer** Judul Ciptaan Aplikasi Learning Programming (LP) 3 Mei 2020, di Jakarta Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia Jangka waktu pelindungan Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama : kali dilakukan Pengumuman. 000186564 Nomor pencatatan : adalah benar berdasarkan keterangan yang diberikan oleh Pemohon. Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014



tentang Hak Cipta.

a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL

> Dr. Freddy Harris, S.H., LL.M., ACCS. NIP. 196611181994031001

# LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Yuyun Yuningsih	Cluster Griya Adipura Sawangan Depok, Blok A3 RT. 002 RW. 004 Kel.Pengasinan Kec. Sawangan
2	Lusa Indah Prahartiwi	Perumahan Sahara Indah Permai 2 Blok D1 No. 1 RT. 7 Rw. 3 Kel. Satria Jaya Kec. Tambun Utara
3	Wulan Dari	JI. Madya No. 6 RT.003 RW. 003 Kel. Tugu Utara Kec. Koja
4	Syarif Hidayatulloh	Jl. Gotong Royong RT.03 RW.02 No.60 Kel.Pondok Bambu Kec.Duren Sawit

## LAMPIRAN PEMEGANG

No	Nama	Alamat
1	Yuyun Yuningsih	Cluster Griya Adipura Sawangan Depok, Blok A3 RT. 002 RW. 004 Kel.Pengasinan Kec. Sawangan
2	Lusa Indah Prahartiwi	Perumahan Sahara Indah Permai 2 Blok D1 No. 1 RT. 7 Rw. 3 Kel. Satria Jaya Kec. Tambun Utara
3	Wulan Dari	Jl. Madya No. 6 RT.003 RW. 003 Kel. Tugu Utara Kec. Koja
4	Syarif Hidayatulloh	Jl. Gotong Royong RT.03 RW.02 No.60 Kel.Pondok Bambu Kec.Duren Sawit





**Aplikasi Learn Programming (LP)** 

Di susun Oleh:

Yuyun Yuningsih

Lusa indah prahartiwi

Wulan Dari

Syarif Hidayatulloh

Manual Book ini menjabarkan bagaimana penggunaan secara detail dari Aplikasi Android mengenai Learn Programming. Aplikasi *Learn Programming* merupakan aplikasi pembelajaran yang membahas tentang Bahasa pemrograman yang sering dipakai di era sekarang. Pada aplikasi ini kami akan menyediakan beberapa Bahasa yang akan dibahas. *Learn programming* akan membahas beberapa Bahasa pemrograman yang terdiri dari pengertian, Struktur pemrograman, Varaible and Operators, Compound Types, Input and Output, Flow of Control, Functions, Object Oriented Approach, Files, Advance Concepts. Di setiap layanan akan kami ringkas dengan pembahasan yang mudah untuk dimengerti dan di berikan contoh untuk setiap Bahasa, sehingga para pembaca bisa untuk langsung mempraktekkan.

Menu yang terdapat pada aplikasi ini adalah :

- 1. Menu Halaman Utama
- 2. Menu Pembelajaran
- 3. Menu Share
- 4. Menu Setting

Lebih jelasnya penulis akan menjabarkan menu dan tampilan serta penggunaannya dari masingmasing menu.Sebagai Contoh penggunaan Aplikasi ini, penulis menjabarkan salah satru menu pembelajaran yaitu menu Pembelajaran Java. Karena untuk penggunaan menu pembelajaran lainnya sama saja dengan menu pembelajaran yang dicontohkan.

#### A. Memulai Program

Untuk memulai program pengguna langsung saja membuka aplikasi Learn Programming (LP).

## B. Tampilan Halaman Utama

Pada Tampilan Utama Berfungsi untuk menampilkan konten halaman utama dari aplikasi yang berupa kumpulan menu Pembelajaran yang ada pada aplikasi Learn Programming (LP). Tampilan Halaman Utama terlihat pada Gambar 1.



Gambar 1. Tampilan Halaman Utama

Pada Tampilan Menu Utama Terdapat beberapa pilihan Menu pembelajaran, *Share*, dan *Setting*, serta *About Learn Proggramming* yang pengguna dapat pilih sesuai yang diinginkan. Pilihan Menu utama sudah tertera pada saat pegguna sudah masuk aplikasi Learn Programming (LP). Tampilan menu utama terlihat pada Gambar 2.



Gambar 2. Tampilan Halaman Menu Utama

# C. Menu Pembelajaran Java

Sebagai contoh disini penulis memilih pembelajaran java, bagaimana penggunaan Aplikasi pembelajaran Java. Pada Menu Pembelajaran pengguna dapat memilih materi tentang java programming yang ingin dipelajari pada sub-sub menu materi Pembelajaran sub-sub materi pembelajaran java dapat dilihat pada Gambar 3.



Gambar 3. Sub-sub Menu Materi pembelajaran Java

## 1) Sub menu Introduction Pembelajaran Java

Pada sub menu *Introduction* menampilkan penjelasan tentang apa itu Pemrograman java. Tampilan Sub menu *intorduction* dapat dilihat pada gambar 4.

moudotion	
Apa itu Java?	
Java adalah baha	sa pemrograman yang
popular, dibuat pao	da tahun 1995.
Java dimiliki oleh	Oracle, dan lebih dari 3 miliar
perangkat menjala	nkan Java.
Digunakan unt	uk:
1. Aplikasi seluler	(khususnya aplikasi Android)
2. Aplikasi deskto	p
<ol><li>Aplikasi web</li></ol>	
4. Server web dan	server aplikasi
5. Games	
6. Koneksi basis c	lata
Dan masin banyar	ciagi
Mongana Mon	guunakan Jaya?
1 Java berfungsi	pada berbagai platform
(Windows Mac Li	nux. Raspberry Pi, dll.
2. Ini adalah salah	satu bahasa pemrograman
paling popular di d	lunia
3. Mudah dipelaja	ri dan mudah digunakan
4. open-source da	in gratis
5. aman, cepat da	n kuat
6. memiliki dukun	gan komunitas yang sangat
besar (puluhan juti	a pengembang)
Karena Java deka	t dengan C ++ dan C #, itu
memudahkan prog	jrammer untuk beralih ke Java

Gambar 4. Sub Menu Introduction Pembelajaran Java

#### 2) Sub Menu Setup & Basic Program

Pada Menu *Setup dan Basic Program* ini menampilkan penjelasan bagaimana cara menginstal pemrograman java, apa saja yang harus diinstal untuk pemrograman java. Tampilan *Setup & Basic Program* terlihat pada Gambar 5.



Gambar 5. Sub Menu Setup & Basic Pembelajaran Java

#### 3) Sub Menu Variable & Data Types

Pada sub menu ini menampilkan penjelasan tentang apa itu *Variable dan Data types* yang digunakan dalam pemrograman java. Tampilan Sub menu *Variable & data types* terlihat pada Gambar 6.

#### Variables & Data Types Variables & Data Types ketika kelas dimuat dalam memori. Variabel Java Contoh untuk memahami jenis variabel di java Variabel adalah wadah yang menyimpan kelas A{ nilai saat program java dijalankan. Variabel int data = 50; // variabel instan ditugaskan dengan tipe data. Variabel adalah static int m = 100; // variabel statis nama lokasi memori. Ada tiga jenis variabel di metode batal () { java: local, instance dan static. int n = 90; // variabel lokal Jenis Variabel } Ada tiga jenis variabel di java: - variabel lokal - variabel intstans Tipe Data di Java - variabel statis Tipe data menentukan berbagai ukuran dan nilai Jenis variabel dalam java yang dapat disimpan dalam variabel. Ada dua 1) Variabel Lokal jenis tipe data di Java: Variabel yang dideklarasikan di dalam tubuh Tipe data primitif: Tipe data primitif meliputi metode disebut variabel lokal. Anda dapat boolean, char, byte, pendek, int, panjang, float, menggunakan variabel ini hanya dalam metode dan double. itu dan metode lain di kelas bahkan tidak Tipe data non-primitif: Tipe data non-primitif menyadari bahwa variabel itu ada. meliputi Kelas, Antarmuka, dan Array. Variabel lokal tidak dapat didefinisikan dengan Tipe Data Primitif Java kata kunci "statis". Dalam bahasa Java, tipe data primitif adalah 2) Variabel Instans blok penyusun manipulasi data. Ini adalah tipe Variabel yang dideklarasikan di dalam kelas data paling dasar yang tersedia dalam bahasa tetapi di luar tubuh metode, disebut variabel Java. instan. Itu tidak dinyatakan sebagai statis. Java adalah bahasa pemrograman yang diketik disebut variabel instan karena nilainya spesifik secara statis. Artinya, semua variabel harus instance dan tidak dibagi di antara instance. dideklarasikan sebelum digunakan. Itu sebabnya 3) Variabel Statis kita perlu mendeklarasikan jenis dan nama Variabel yang dinyatakan sebagai statis disebut variabel. variabel statis. Itu tidak bisa lokal. Anda dapat Data Type Default Value Default size membuat satu salinan variabel statis dan boolean false 1 bit berbagi di antara semua instance kelas. Alokasi char ri untuk variahal etatic hanva tariadi cakali

# Gambar 6. Sub Menu Variable & Data Types Pembelajaran Java

## 4) Sub Menu Operator & Expressions

Pada sub menu ini menampilkan penjelasan tentang apa itu *Operator & Expressions* yang digunakan dalam pemrograman java. Tampilan Sub menu *Operator & Expressions* terlihat pada Gambar 7.



Gambar 7. Sub Menu Variable & Data Types Pembelajaran Java

## 5) Sub Menu Flow Control

Pada sub menu ini menampilkan penjelasan tentang apa *itu Flow Control* dalam pemrograman java. *Flow Control* adalah membuat pilihan pada kode kita agar kita bisa menjalankan baris lain pada kode didalam kondisi atau kendala tertentu, *Flow control* terdiri dari Percabangan (*Branching*) dan Perulangan (*Looping*). Tampilan Sub Menu *Flow Control* terlihat pada Gambar 8.



Gambar 8. Tampilan Sub Menu Control Flow

Dalam beberapa menu pembelajaran terdapat sub menu juga di dalamnya contohnya untuk menu pembelajaran *flow control* ini, namun ada pula untuk menu yang lain seperti Percabangan dan Perulangan. Yang termasuk dari percabangan adalah Sub Menu *If Statement* dan contoh penggunaannya yang terlihat pada Gambar 9.

#### If Statement

#### Kondisi Java dan Pernyataan If

Java mendukung kondisi logis yang biasa dari matematika: Kurang dari: a <b

Kurang dari atau sama dengan <= b

Lebih besar dari: a > b

Lebih besar dari atau sama dengan: a >= b Sama dengan a == b

Tidak Setara dengan: a != B

Anda dapat menggunakan kondisi ini untuk melakukan tindakan berbeda untuk keputusan yang berbeda.

Java memiliki pernyataan kondisional berikut: - Gunakan jika untuk menentukan blok kode yang akan dieksekusi, jika kondisi yang ditentukan benar

- Gunakan yang lain untuk menentukan blok kode yang akan dieksekusi, jika kondisi yang sama salah

- Gunakan lagi jika menentukan kondisi baru untuk diuji, jika kondisi pertama salah

- Gunakan sakelar untuk menentukan banyak blok kode alternatif yang akan dieksekusi

#### Pernyataan if

Gunakan pernyataan if untuk menentukan blok kode Java yang akan dieksekusi jika kondisinya benar. Syntax

if (kondisi) { // blak koda yang akan diakaakwai jika

#### **If Statement**

bahwa x lebih besar dari y. Pernyataan lain Gunakan pernyataan lain untuk menentukan blok kode yang akan dieksekusi jika kondisinya salah. Syntax if (kondisi) { // blok kode yang akan dieksekusi jika kondisinya benar } lain { // blok kode yang akan dieksekusi jika kondisinya salah Contoh waktu int = 20; if (waktu <18) { System.out.println (Selamat siang.); } lain { System.out.println (Selamat malam.); // Keluaran Selamat malam. Contoh dijelaskan Pada contoh di atas, waktu (20) lebih besar dari 18, sehingga kondisinya salah. Karena itu, kami beralih ke kondisi lain dan mencetak ke lavar Selamat malam. Jika waktunya kurang dari 18, program akan mencetak Selamat siang. Lain lagi kalau Statement Gunakan pernyataan else if untuk menentukan

# kondisi baru jika kondisi pertama salah.

If Statement

Syntax if (condition1) { // blok kode yang akan dieksekusi jika kondisi1 benar } lain jika (kondisi2) { // blok kode yang akan dieksekusi jika kondisi1 salah dan kondisi2 benar } lain { // blok kode yang akan dieksekusi jika kondisi1 salah dan kondisi2 salah Contoh waktu int = 22; if (waktu <10) { System.out.println (Selamat pagi.); } lain jika (waktu <20) { System.out.println (Selamat siang.); } lain { System.out.println (Selamat malam.); // Keluaran Selamat malam. Contoh dijelaskan Dalam contoh di atas, waktu (22) lebih besar dari 10, jadi kondisi pertama salah. Kondisi berikutnya, di pernyataan if-if lainnya, juga false, jadi beralih ke kondisi lain karena condition1 dan condition2 keduanya false - dan mencetak ke lavar Selamat malam. Namun, jika waktunya 14, program kami akan mencetak Selamat siang. Short Hand If ... Else (Operator Ternary) Ada juga short-hand if else, yang dikenal

#### **If Statement**

avar selamat malam. Namun, jika waktunya 14, program kami akan mencetak Selamat siang. Short Hand If ... Else (Operator Ternary) Ada juga short-hand if else, yang dikenal sebagai operator ternary karena terdiri dari tiga operan. Itu dapat digunakan untuk mengganti beberapa baris kode dengan satu baris. Ini sering digunakan untuk menggantikan pernyataan sederhana jika lain: Syntax variabel = (kondisi)? expressionTrue: expressionFalse; Alih-alih menulis: Contoh waktu int = 20; if (waktu <18) { System.out.println (Selamat siang.); } lain { System.out.println (Selamat malam.); Anda cukup menulis: Contoh waktu int = 20; Hasil string = (waktu <18)? Selamat siang. : Selamat malam System.out.println (hasil); Kirim masukan Histori Disimpan Komunitas

#### Gambar 9. Tampilan Sub Menu IF Statement

#### **If Statement**

Syntax if (kondisi) { // blok kode yang akan dieksekusi jika kondisinya benar Perhatikan bahwa jika dalam huruf kecil. Huruf besar (Jika atau IF) akan menghasilkan kesalahan. Dalam contoh di bawah ini, kami menguji dua nilai untuk mengetahui apakah 20 lebih besar dari 18. Jika kondisinya benar, cetak beberapa teks: Contoh if (20 >18) { System.out.println (20 lebih besar dari 18); Kami juga dapat menguji variabel: Contoh int x = 20; int y = 18; if (x > y) {

System.out.println (x lebih besar dari y);

#### Contoh dijelaskan

}

Dalam contoh di atas kita menggunakan dua variabel, x dan y, untuk menguji apakah x lebih besar dari y (menggunakan operator>). Karena x adalah 20, dan y adalah 18, dan kita tahu bahwa 20 lebih besar dari 18, kita mencetak ke layar Selain dari IF *statement* yang termasuk Percabangan, salah satu yang termasuk dari percabangan adalah *Switch Statement*. Sub *Switch* Statement ini menjelaskan tentang apa itu *Switch Statement* dan contoh penggunaannya pada Pemrograman Java terlihat pada gambar 10.

	7.47 必见金足上 📟	747 원 전 જ 내 또 📟
Switch Statement	Switch Statement	Switch Statement
Pernyataan Switch Java Gunakan pernyataan beralih untuk memilih salah satu dari banyak blok kode yang akan dieksekusi. Syntax switch (ekspresi) { kasus x: // blok kode istirahat; kasus y: // blok kode istirahat; default: // blok kode } Begini Cara kerjanya: - Ekspresi sakelar dievaluasi sekali. - Nilai ekspresi dibandingkan dengan nilai setiap kasus. - Jika da kecocokan, blok kode terkait dijalankan. Contoh di bawah ini menggunakan nomor hari kerja untuk menghitung nama hari kerja: Contoh: hari int = 4; switch (hari) { case 1: System.out.println (Monday); break; one 2:	Case 2. System.out.println (Tuesday); break; case 3: System.out.println (Wednesday); break; case 4: System.out.println (Thursday); break; case 5: System.out.println (Friday); break; } // Keluaran Thursday (hari 4) Kata kunci istirahat Ketika Java mencapai kata kunci istirahat, itu keluar dari blok saklar. Ini akan menghentikan pelaksanaan lebih banyak kode dan pengujian case di dalam blok. Ketika keccockan ditemukan, dan pekerjaan selesai, saatnya istirahat. Tidak perlu pengujian lebih lanjut. Istirahat dapat menghemat banyak waktu eksekusi karena mengabaikan eksekusi semua kode lainnya di blok saklar. Kata Kunci default Kata kunci default Kata kunci default Contoh hari int = 4; switch (hari) {	System.out.prinun (+nday); break; } // Keluaran Thursday (hari 4) Kata kunci istirahat Ketika Java mencapai kata kunci istirahat, itu keluar dari blok saklar. Ini akan menghentikan pelaksanaan lebih banyak kode dan pengujian case di dalam blok. Ketika kecocokan ditemukan, dan pekerjaan selesai, saatnya istirahat. Tidak perlu pengujian lebih lanjut. Istirahat dapat menghemat banyak waktu eksekusi karena mengabaikan eksekusi semua kode lainnya di blok saklar. Kata Kunci default Kata kunci default Kata kunci default Kata kunci default Contoh hari int = 4; switch (hari) { case 6: System.out.println (Today is Saturday); break; case 7: System.out.println (Today is Sunday); break; default: System.out.println (Menantikan Akhir Pekan); } // Keluaran Menantikan Akhir Pekan

#### Gambar 10. Tampilan Sub Menu Switch Statement

Selain dari percabangan yang termasuk dalam *Flow Control*, Perulangan juga termasuk dari *Flow control* yang digunakan dalam pemrograman java. *Looping* yang termasuk kedalam *flow control* adalah While, Do While dan For. Sub menu While, Do While dan For Loop dan contoh penggunaannya terlihat dalam Gambar 11.

While Loop

Java While Loop

LOOD Loop dapat menjalankan blok kode selama kondisi yang ditentukan tercapai. Loop berguna karena menghemat waktu, mengurangi kesalahan, dan membuat kode lebih mudah dibaca. Java While Loop Loop sementara melalui blok kode loop selama kondisi yang ditentukan benar: Syntax while (kondisi) { // blok kode yang akan dieksekusi Pada contoh di bawah ini, kode dalam loop akan berjalan, berulang-ulang, selama variabel (i) kurang dari 5: Contoh int i = 0;while (i <5) { System.out.println (i); i ++: } Catatan: Jangan lupa untuk menambah variabel yang digunakan dalam kondisi tersebut, jika tidak, loop tidak akan pernah berakhir!

#### For Loop

Conton dijelaskan

Pernyataan 1 menetapkan variabel sebelum loop dimulai (int i = 0). Pernyataan 2 mendefinisikan kondisi untuk menjalankan loop (saya harus kurang dari 5). Jika kondisinya benar, loop akan memulai lagi, jika itu salah, loop akan berakhir. Pernyataan 3 meningkatkan nilai (i ++) setiap kali blok kode dalam loop telah dieksekusi. Contoh lain Contoh ini hanya akan mencetak nilai genap antara 0 dan 10: untuk (int i = 0; i <= 10; i = i + 2) { System.out.println (i); Untuk-Setiap Loop Ada juga loop untuk masing-masing, yang digunakan secara eksklusif untuk mengulang elemen-elemen dalam array: Syntax for (type variableName: arrayName) { // blok kode yang akan dieksekusi Contoh berikut menampilkan semua elemen dalam larik mobil, menggunakan loop untuk masing-masing: String[] mobil = {Volvo, BMW, Ford, Mazda}; for (String i: cars) { System.out.println (i);

**Do While loop** The Do / While Loop Do / while loop adalah varian dari while loop. Loop ini akan mengeksekusi blok kode sekali, sebelum memeriksa apakah kondisinya benar, maka itu akan mengulang loop selama kondisinya benar. Syntax do // blok kode yang akan dieksekusi while(kondisi); Contoh di bawah ini menggunakan do / while loop. Loop akan selalu dieksekusi setidaknya sekali, walaupun kondisinya salah, karena blok kode dieksekusi sebelum kondisi diuji: Contoh int i = 0;do { System.out.println (i); i ++; while (i <5); For Loop Java For Loop Ketika Anda tahu persis berapa kali Anda ingin mengulang melalui blok kode, gunakan loop untuk bukan loop sementara: Syntax



Gambar 11. Tampilan Sub Menu While, Do While dan For Loop

#### 6) Sub Menu Classes, Object & Method

Sub menu ini menjelaskan tentang *Classes, Object & Method* pada pemrograman java. Dalam menu *classes, object & methods* terdapat sub-menu pembelajrannya yang dapat dilihat pada Gambar 12.



Gambar 12. Tampilan Menu-menu yang Terdapat Pada Sub Menu Classes, Object & Method

Adapun Penjelasan Tentang Sub Menu yang terdapat pada Sub menu *Classes, Object & Method* adalah sebagai berikut:

a. Sub menu Classes Basic

Menu Ini menjelaskan tentang *Classes Basic* yang digunakan pada pemrograman java, dan tampilan sub menu ini dapat dilihat pada Gambar 13.



Gambar 12. Sub Menu Classes Basic

b. Class Objects

Menu Ini menjelaskan tentang *Classes Objects* yang digunakan pada pemrograman java, dan tampilan sub menu ini dapat dilihat pada Gambar 13.

Class Objects		
	-	
Kelas Obyek adal	ı ah kelas induk dari semua	
kelas di java secara	default. Dengan kata lain, ini	
adalah kelas paling	atas dari java.	
Kelas Obyek bern	nanfaat jika Anda ingin	
merujuk objek apa	pun yang jenisnya tidak Anda	
ketahui. Perhatikan	bahwa variabel referensi	
kelas induk dapat n	nerujuk objek kelas anak,	
yang dikenal sebag	ai upcasting.	
() yang mengemba	likan objek tetani bisa dari	
lenis apa sala sepe	rti Karvawan. Mahasiswa dil.	
kita bisa mengguna	akan referensi kelas Objek	
untuk merujuk obje	k itu. Sebagai contoh:	
Object obj = get	Object (); // kita tidak tahu	
objek apa yang aka	n dikembalikan dari metode	
ini Kalaa Ohiaataa	and a low as the share as a soll a loss	
Kelas Object meny	ediakan beberapa perilaku	
dibandingkan obje	k dapat dikloping objek dapat	
dapat diberitahu dil		
Kelas Object meny	ediakan banyak metoda	
Mereka adalah seb	agai berikut:	
Deskripsi Metode		
- public getClass a	khir kelas () mengembalikan	
objek kelas Class d	ari objek ini. Kelas	
Kelas selanjutnya d	apat digunakan untuk	
mendapatkan meta	data dari kelas ini.	
- public int hashCo	de () mengembalikan nomor	

Gambar 13. Sub Menu Classes Basic

#### c. Constructor

Menu Ini menjelaskan tentang *Constructor* yang digunakan pada pemrograman java, dan tampilan sub menu ini dapat dilihat pada Gambar 14.

opotruotoro	
onstructors	
Apa itu Java?	
Java adalah bahas	a pemrograman yang
Java dimiliki oleh	a tanun 1995. Oraele, dan lehih dari 2 miliar
Java diriliki oleh o	skap Jova
Serangkat menjalah	ikan Java.
Digunakan untu	ık.
1. Aplikasi seluler	(khususnva aplikasi Android)
2. Aplikasi desktor	5
3. Aplikasi web	
4. Server web dan	server aplikasi
5. Games	
6. Koneksi basis di	ata
Dan masih banyak	lagi
Mengapa Meng	ngunakan Jaya?
1. Java berfungsi r	pada berbagai platform
Windows, Mac, Lir	ux, Raspberry Pi, dll.
2. Ini adalah salah	satu bahasa pemrograman
baling popular di di	unia
3. Mudah dipelajar	'i dan mudah digunakan
<ol><li>open-source dar</li></ol>	n gratis
5. aman, cepat dar	n kuat
<ol><li>memiliki dukung</li></ol>	jan komunitas yang sangat
oesar (puluhan juta	pengembang)
Karena Java dekat	dengan C ++ dan C #, itu
nemudahkan prog	rammer untuk beralih ke Java
atau sebaliknya	

Gambar 14. Sub Menu Constructor

d. Method Overloading

Menu Ini menjelaskan tentang Method Overloading yang digunakan pada pemrograman

java, dan tampilan sub menu ini dapat dilihat pada Gambar 15.



Gambar 15. Sub Menu Method Overloading

#### e. Static Keyword

Menu Ini menjelaskan tentang *Static Keyword* yang digunakan pada pemrograman java, dan tampilan sub menu ini dapat dilihat pada Gambar 16.

Gambar 16. Sub Menu Static Keyword

f. Inheritance

Menu Ini menjelaskan tentang *Inheritance* yang digunakan pada pemrograman java, dan tampilan sub menu ini dapat dilihat pada Gambar 17.



Gambar 17. Sub Inheritance

#### g. Types Of Heritance

Menu Ini menjelaskan tentang *Types Of Heritance* yang digunakan pada pemrograman java, dan tampilan sub menu ini dapat dilihat pada Gambar 18.



Gambar 18. Sub Menu Types Of Heritance

#### h. Final Keyword

Menu Ini menjelaskan tentang *Final Keyword* yang digunakan pada pemrograman java, dan tampilan sub menu ini dapat dilihat pada Gambar 19.



Gambar 19. Sub Menu Final Keyword

#### i. Abstraction

Menu Ini menjelaskan tentang *Abstraction* yang digunakan pada pemrograman java, dan tampilan sub menu ini dapat dilihat pada Gambar 20.



Gambar 20. Sub Menu Abstraction

## 7) Sub Menu Arrays, String & Vector

Pada sub menu ini menampilkan penjelasan tentang apa itu Arrays, String & Vector dalam pemrograman java. Tampilan sub menu Arrays, String & Vector dapat dilihat pada gambar 21. Dan sub menu Arrays, String & Vector juga memiliki sub-sub menu lainnya yang berisi tentang penjelasan dari Arrays, String & Vector serta Wrap Class.



Gambar 21. Sub Menu Arrays, String & Vector

## 8) Sub Menu Interfaces

Pada sub menu ini menampilkan penjelasan tentang apa *Interfaces* dalam pemrograman java, dan *Interface* memiliki beberapa bagian yaitu *Define Interfaces, Extending Interfaces, dan Implementasi interfaces.* Tampilan sub menu Define Interfaces dapat dilihat pada Gambar 22-24.

Define Interfaces			
Cara lain untuk meno adalah dengan interfa	i capai abstraksi di Java, ces		
Antarmuka(Interfaces	s) adalah kelas		
abstrak yang sepenuh	nya digunakan untuk		
mengelompokkan me kosong:	tode terkait dengan benda		
interface Animal (			
public void animalSo (does not have a body	und(); // interface method		
public void run(); // in have a body)	terface method (does not		
}			
untuk mengakses met	tode antarmuka,		
antarmuka harus "diim	nplementasikan" (agak		
kunci implement (alib	en kelas lain dengan kata -alih meluas) Isi metode		
antarmuka disediakan // Interface	oleh kelas "implement":		
interface Animal {			
public void animalSo (does not have a body	und(); // interface method /)		
public void sleep(); // not have a body)	interface method (does		
}			
// Pig implements the	e Animal interface		
class Pig implements	s Animal {		
public void animalSo	und() {		

Gambar 22. Define Interfaces

Extended Interfaces Antarmuka dapat memperluas antarmuka lain dengan cara yang sama seperti kelas dapat memperluas kelas lain. Kata kunci extends digunakan untuk memperluas antarmuka, dan antarmuka anak mewarisi metode antarmuka induk. Antarmuka Olahraga berikut diperpanjang oleh antarmuka Hoki dan Sepak Bola. Example // Filename: Sports.java public interface Sports { public void setHomeTeam(String name); public void setVisitingTeam(String name); // Filename: Football.java public interface Football extends Sports { public void homeTeamScored(int points); public void visitingTeamScored(int points); public void endOfQuarter(int quarter); // Filename: Hockey.java public interface Hockey extends Sports { public void homeGoalScored(); public void visitingGoalScored(); public void endOfPeriod(int period); public void overtimePeriod(int ot);

Antarmuka Hoki momiliki omnat motodo

#### Gambar 23. Extending Interfaces

#### Implementing Interfaces

Implementing Interface Untuk mendeklarasikan kelas yang mengimplementasikan antarmuka, Anda menyertakan klausa implementasi dalam deklarasi kelas. Kelas Anda dapat mengimplementasikan lebih dari satu antarmuka, sehingga kata kunci implement diikuti oleh daftar antarmuka yang dipisahkan oleh koma yang diterapkan oleh kelas. Dengan konvensi, klausa implement mengikuti klausa extends, jika ada.

Antarmuka Sampel, Relatable Pertimbangkan antarmuka yang mendefinisikan cara membandingkan ukuran objek.

public interface Relatable { // this (object calling isLargerThan) // and other must be instances of

// the same class returns 1, 0, -1
// if this is greater than,
// equal to, or less than other
public int isLargerThan(Relatable other);

Jika Anda ingin dapat membandingkan ukuran objek yang sama, apa pun itu, kelas yang instantiate mereka harus mengimplementasikan Relatable.

Setiap kelas dapat mengimplementasikan Relatable jika ada beberapa cara untuk membandingkan ukuran relatif dari objek yang dibuat dari kelas. Untuk string, bisa jadi jumlah karakter; untuk buku, bisa jadi jumlah n-untuk ciewa itu hica ha -+- 1

**Gambar 24.** Implementing Interfaces

#### 9) Sub Menu Package

Pada sub menu ini menampilkan penjelasan tentang apa *Package* dalam pemrograman java. Tampilan sub menu Package dapat dilihat pada Gambar 25.

2:53	
Package	
INTRODUCTION	TO PACKAGES
	DACKAGE
JAVA APT	PACKAGE
SYSTEM P	ACKAGES

## Gambar 25. Tampilan Sub Menu Package

Pada sub menu Package ada beberapa menu-menu yang dapat pengguna gunakan untuk dapat lebih mengenal lagi package. Menu-menu tersebut adalah:

a. Introduction to Package

Menu ini menampilkan penjelasan tentang apakah itu *Introduction Package*, dan dapat dilihat pada gambar 26.



Gambar 26. Introduction to Package

b. Java API Package

Menu ini menampilkan penjelasan tentang apakah itu Java API *Package*, dan dapat dilihat pada gambar 27.



Gambar 27. Java API Package

c. System Package

Menu ini menampilkan penjelasan tentang apakah *itu System Package*, dan dapat dilihat pada gambar 28.



Gambar 28. System Package

#### 10) Sub Menu Multithread Programming

Pada sub menu ini menampilkan penjelasan tentang apa *Multithread Programming* dalam pemrograman java. Tampilan sub menu *Multithread Programming* dapat dilihat pada Gambar 29.



Gambar 29. Tampilan Sub Menu Multithread Programming

Pada sub menu *Multithread Programming* ada beberapa menu-menu yang dapat pengguna gunakan untuk dapat lebih mengenal lagi package. Menu-menu tersebut adalah:

a. Create Thread

Menu ini menampilkan penjelasan tentang apakah itu *Create Thread*, dan dapat dilihat pada Gambar 30.



Gambar 30. Tampilan Create Thread

b. Stopping & Blocking

Menu ini menampilkan penjelasan tentang apakah itu *Stopping & Blocking*, dan dapat dilihat pada Gambar 31.



Gambar 31. Tampilan Stopping & Blocking

c. Life Cycle

Menu ini menampilkan penjelasan tentang apakah *itu Life Cycle*, dan dapat dilihat pada Gambar 32.



Gambar 32. Tampilan Life Cycle

d. Exception & Priority

Menu ini menampilkan penjelasan tentang apakah itu *Exception & Priority*, dan dapat dilihat pada Gambar 33.

Exception & Priority	Exception & Priority	
Exception IllegalArgumentException jika prioritasnya tidak dalam kisaran MIN_PRIORITY hingga MAX_PRIORITY. SecurityException jika thread saat ini tidak dapat mengubah thread ini. Priority of a Thread (Thread Priority): Java mengizinkan kami untuk mengatur prioritas utas menggunakan metode setPriority() sebagai berikut: ThreadName.setPriority (intNumber); Setiap utas memiliki prioritas. Prioritas diwakili oleh angka antara 1 dan 10. Dalam kebanyakan kasus, thread schedular menjadwalkan thread sesuai dengan prioritasnya (dikenal sebagai preemptive schedulling). tetapi tidak dijamin karena itu tergantung pada spesifikasi JVM apa yang dipilih schedulling itu. 3 konstanta defiend di kelas Thread: public static int MIN_PRIORITY public static	Exception IllegalArgumentException jika prioritasnya tidak dalam kisaran MIN_PRIORITY hingga MAX_PRIORITY. SecurityException jika thread saat ini tidak dapat mengubah thread ini. Priority of a Thread (Thread Priority): Java mengizinkan kami untuk mengatur prioritas utas menggunakan metode setPriority) sebagai berikut: ThreadName.setPriority (intNumber): Setiap utas memiliki prioritas. Prioritas diwakili oleh angka antara 1 dan 10. Dalam kebanyakan kasus, thread schedular menjadwalkan thread sesuai dengan prioritasnya (dikenal sebagai preemptive schedulling). tetapi tidak dijamin karena itu tergantung pada spesifikasi JVM apa yang dipilih schedulling itu. 3 konstanta defiend di kelas Thread: public static int NIN_PRIORITY public static int MAX_PRIORITY Prioritas default thread adalah 5 (NORM_PRIORITY) Nilai MIN_PRIORITY adalah 1. Nilai MAX_PRIORITY adalah 1. Nilai MAX_PRIORITY adalah 10. class TestPriority extends Thread{ public vid run(){	

Gambar 33. Tampilan Exception & Priority

e. Synchronization

Menu ini menampilkan penjelasan tentang apakah itu *Synchronization*, dan dapat dilihat pada Gambar 34.



Gambar 34. Tampilan Synchronization

f. Runnable Interface

Menu ini menampilkan penjelasan tentang apakah *itu Runnable Interface*, dan dapat dilihat pada Gambar 35.



Gambar 35. Tampilan Runnable Interface

## 11) Sub Menu Error & Exception

Pada sub menu ini menampilkan penjelasan tentang apa *Error & Exception* dalam pemrograman java. Tampilan sub *menu Error & Exception* dapat dilihat pada Gambar 36.



Gambar 35. Tampilan Sub Menu Error & Exception

## D. Menu Sharing

Pada Menu ini menampilkan penjelasan tentang apa *Sharing* materi via media sosial dan yang lainnya dalam Aplikasi Learn Programming. Tampilan Menu *Sharing* dapat dilihat pada Gambar 36.



Gambar 36. Tampilan Menu Sharing

# E. Menu Setting

Pada Menu ini menampilkan penjelasan tentang bagaimana mengatur Aplikasi Learn Programming (LP) .Tampilan Menu *Setting* dapat dilihat pada Gambar 37.

Setting	
Referencess	
Tell your friend	
About this App	

Gambar 37. Tampilan Menu Setting

Pada Menu *Setting* Memiliki Sub-Sub menu dalam melakukan pengaturan dari Aplikasi Learn Programming(LP). Sub-sub menu antara lain:

1. Sub Menu *References* merupakan alamat link yang menjadi referensi kami dalam memuat materi dalam aplikasi ini. Tampilan sub menu reference dapat dilihat pada gambar 38.



Gambar 38. Tampilan Sub Menu References

 Sub Menu *Tell Your Friend* sama seperti menu *share* sebelumnya. Tampilan menu Telling Your Friend dapat dilihat pada Gambar 39.

7:58		\$\$ @	n 🕾 🔊				
Setting							
Referencess	Referencess						
Tell your frie	nd						
About this App							
Sharing Option							
$\odot$	$\odot$						
Transfer data-free	WhatsApp	Direct Message	Twee	et			
			E				
Telegram	Copy to clipboard	Save to Drive	Save to Clou	WPS d			
•••							
Cancel							

Gambar 39. Tampilan Sub Menu Tell Your Friend

3. Selanjutnya yang terakhir adalah menu *About this app*, menu ini berisi tentang aplikasi Learn Programming dan pembuat aplikasi ini. Tampilan Sub menu About dapat dilihat tanggal 40.



Gambar 40. Tampilan Sub Menu About Learn Programming

# F. Tutup Aplikasi

Untuk menutup aplikasi Learn Programming, tekan tombol *back* pada perangkat android anda, otomatis Aplikasi Learn Programming tertutup.