

# IEVADS JAVA PROGRAMMĒŠANĀ

Java 1.0



# IEVADLEKCIJA

## Lekcija # 01





## Oskars Linde

**Java Software developer,  
SIA C.T.Co**

- Pieredze – 5+ gadi
- Java – kopš 2017. gada
- Bijušais testētājs
- Pašlaik – Korporatīvie IT risinājumi
- Patīk risināt jaunas, izaicinošas problēmas
- Novērtēju kodu, kas rakstīts cilvēkiem
- Ārpus darba - improvizācijas teātris, grāmatas, šahs



**Java Guru**

- Darbojamies **10+** gadus
- Apmācīti vairāk kā **3000** studenti
- Konstanti paplašinām piedāvāto kursu klāstu
- Palīdzam studentiem darba meklējumos

# KURSA PĀRSKATS

- Ievads par IT, programmēšanu
- Ievads objektorientētajā programmēšanā
- Ievads Java programmēšanas valodā
- Objektorientētas programmēšanas pamatprincipi
- Algoritmi, datu struktūras un kolekcijas
- Vienībtestēšana
- Java streams & functions
- Clean Code pamatprincipi



# MĀCĪBU PLĀNS

- Lekciju valoda – **Latviešu**
- Lekciju materiāli – **Angliski**
- **16 lekcijas**
- Katra lekcija – **3 (3 x 45 min.)** akadēmiskās stundas
- Praktiski piemēri lekcijas laikā
- Mājasdarbi pēc lekcijas - **obligāti**



# MATERIĀLI UN RESURSI

- Kurša materiāli
  - Java Guru Github
- Grāmatas
  - Head First Java (2nd edition) - Bert Bates, Kathy Sierra
  - Grokking Algorithms - Aditya Y. Bhargava
  - Thinking in Java (4th edition) - Bruce Eckel
- Dokumentācija un palīdzība
  - Oracle's Java Tutorial: <https://docs.oracle.com/javase/tutorial/>
  - Q&A: <https://stackoverflow.com/questions/tagged/java>



# DARBA UZSĀKŠANAI

- Kursa materiāli
- Rīki
  - JDK 17: <https://adoptium.net/temurin/releases/?version=17> (Nav obligāti)
  - IntelliJ IDEA Community Edition: <https://www.jetbrains.com/idea/download/>
  - Git: <https://git-scm.com/downloads>
- Sadarbība
  - GitHub: <https://github.com/>
  - WhatsApp: <https://www.whatsapp.com/download/>





# IEVADS

---



START

# Kā uzsākt savu ceļu IT

“Starting a career in IT requires hard work and dedication, but with the right approach, it's possible to build a successful career in this exciting and dynamic field.”



# Kā uzsākt savu ceļu IT

- Izvēlēties nozari (programmatūras izstrāde, datu bāzu pārvaldība, tīklu administrēšana, kiberdrošība utt.)
- Strādāt pie prasmju attīstīšanas (sertifikāti, kursi)
- Izveido savu portfolio
- Kontakti un tīklošanās (konferences, semināri u.c.)
- Prakse
- CV sūtīšana uz dažādiem uzņēmumiem



## CHECKLIST



# Kā uzsākt savu ceļu IT

Daži uzsākšanas stāsti:

- Paša stāsts
- Brālis - FE programmētājs
- Kolēģis - FE programmētājs
- Kolēģis - no finansēm uz testēšanu
- Kolēģis - no pavāra uz testēšanu
- Paziņa - no mežstrādnieka uz testēšanu
- Paziņa - Apple support uz IT komandas vadīšanu



# Kas ir programmētājs?

Programmētājs ir cilvēks, kas specializējas datorprogrammu izstrādē un rakstīšanā. Viņš izmanto dažādus programmēšanas valodas, koda redaktorus, datu struktūras un algoritmus, lai radītu programmas, kas spēj izpildīt konkrētas funkcijas un uzdevumus. Programmētājs strādā arī pie programmatūras projektu plānošanas, analizēšanas, testēšanas un dokumentēšanas.

Kā norit programmētāja darbadiena?





# JAVA VĚSTURE

---

# VĒSTURE

- Green Team (James Gosling, Mike Sheridan, Patrick Naughton)
- Valodas projekts uzsākts **1991.** gada jūnijā
- Sākotnēji projektēta **mazām integrētām sistēmām** elektriskajās ierīcēs
- Sākotnējais nosaukums Greentalk, vēlāk pārsaukta par Oak
- Dažus gadus vēlāk nosaukta par **Java**



# NOSAUKUMS «JAVA»

- Green Team (James Gosling, Mike Sheridan, Patrick Naughton)
  - Attēlot tehnoloģijas būtību
  - **Revolucionāra** un **dinamiska** tehnoloģija
  - **Viegli** izrunājams un **foršs** lietošanā
- Ieteiktie nosaukumi Dynamic, Silk, Jolt, DNA
- Java ir sala, kur **pirmo reizi** tika ražota **kafija**

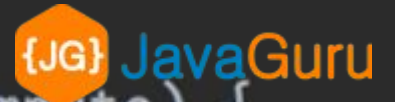




# JAVA PĀRSKATS

---

@Override



```
public Function simplify(boolean compute) {  
    List<Expression> simplifiedOperands = null;  
    for ((int i, Expression operand) : operands)  
        Expression simplifiedOperand = operand.simplify()  
        if (simplifiedOperand != operand) {  
            if (simplifiedOperands == null) {  
                simplifiedOperands = new ArrayList<>();  
                if (i > 0) {  
                    simplifiedOperands.addAll(operands.subList(0, i));  
                }  
            }  
            simplifiedOperands.add(simplifiedOperand);  
        } else {  
            if (simplifiedOperands != null) {  
                simplifiedOperands.add(operand);  
            }  
        }  
    }  
}
```

# KOMPONENTES

## JAVA VIRTUAL MACHINE (JVM)

Abstrakta skaitļošanas mašīna, ar kuras palīdzību dators prot izpildīt **Java** programmu.

## JAVA RUNTIME ENVIRONMENT (JRE)

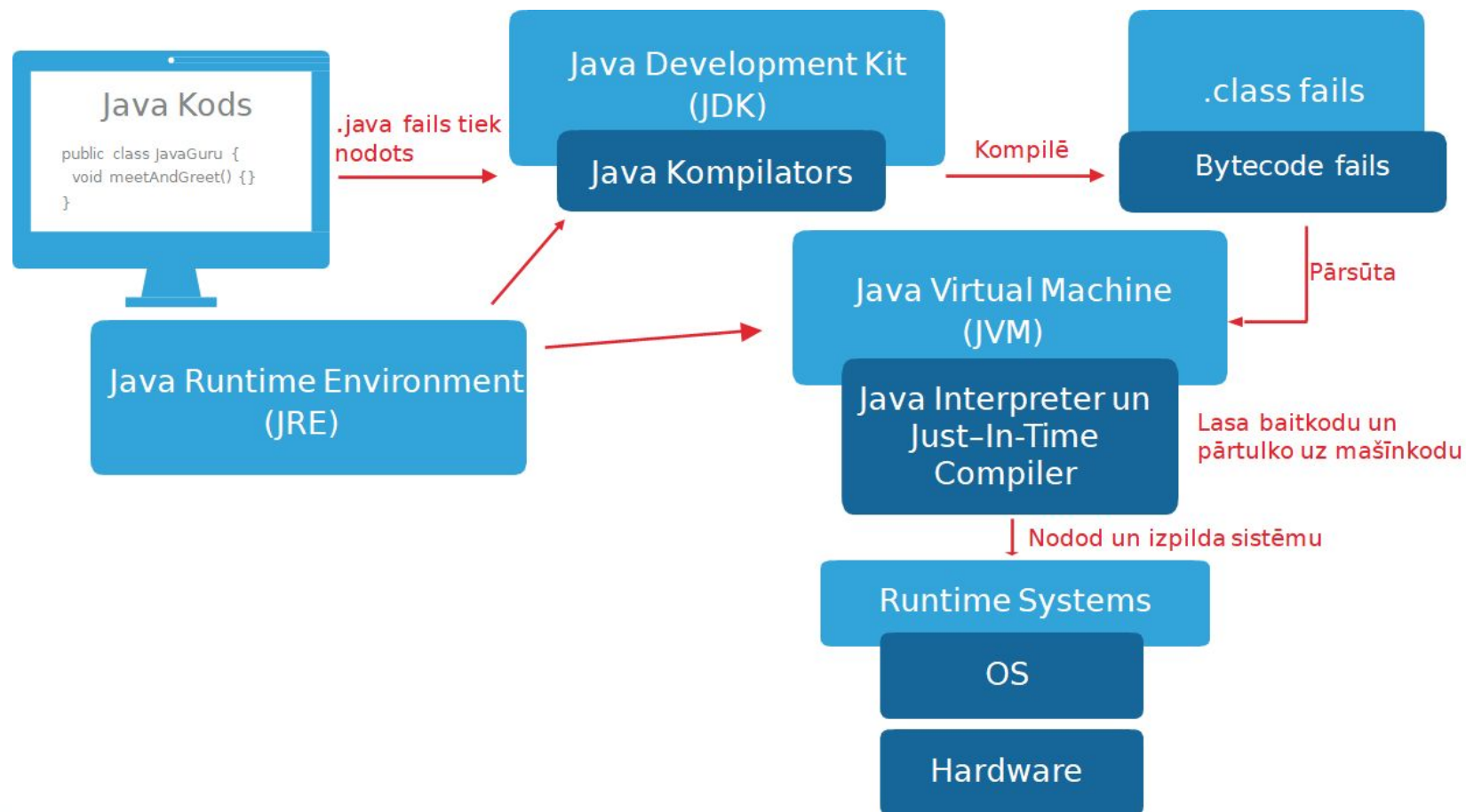
Programmatūras pakotne, kas satur visas nepieciešamās programmas, lai **Java** programmu varētu izpildīt - palaist. Tā satur sevī **JVM** implementāciju kopā ar **Java Class Library (JCL)**.

## JAVA DEVELOPMENT KIT (JDK)

Satur sevī gan **JRE**, gan arī visus nepieciešamos rīkus (programmas) **Java** programmu izstrādei un izpildei.



# INTERNAL WORKFLOW



# PRINCIPI

**Vienkāršība**

**Neatkarība no platformas**

**Augsta veiktspēja**

**Daudzpavedienu**

**Droša**

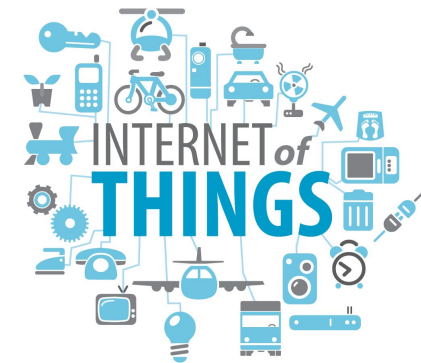
**Objektorientēta**



# JAVA PIELIETOJUMS



**NETFLIX**



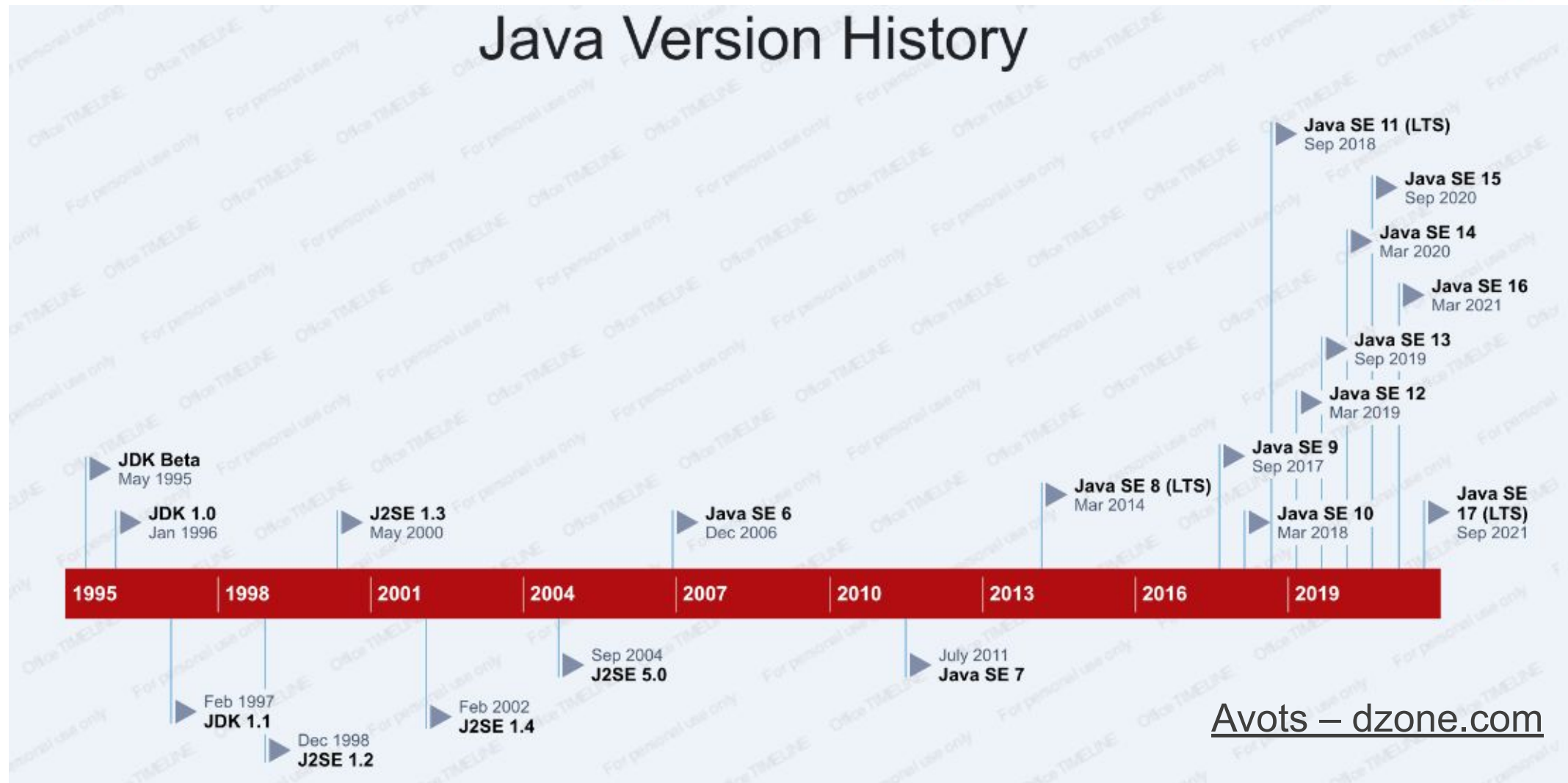


# JAVA LAIDIENU VĒSTURE

---



# Java Version History

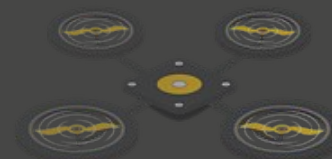


[Avots – dzone.com](https://dzone.com)



# OBJEKT-ORIENTĒTĀ PROGRAMMĒŠANA

---





# OBJEKTI

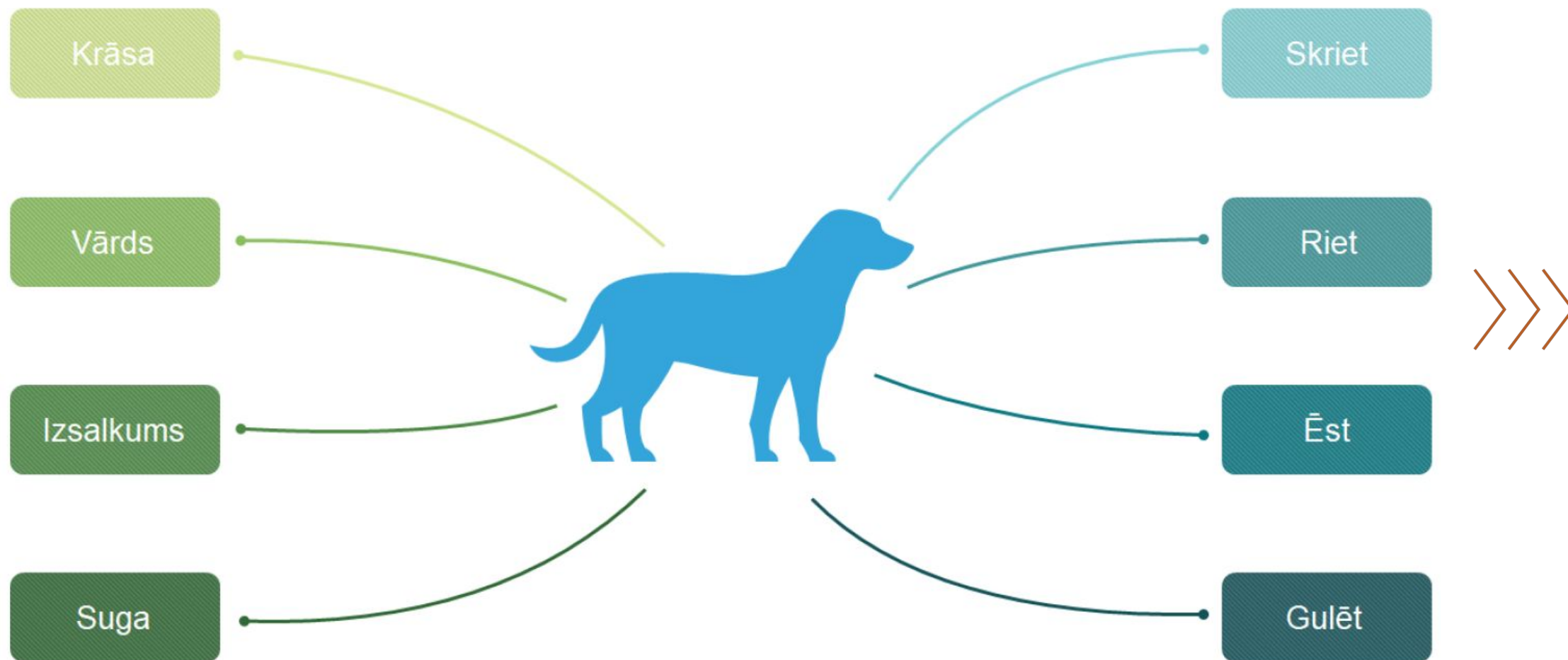


# OBJEKTU RAKSTURLIELUMI

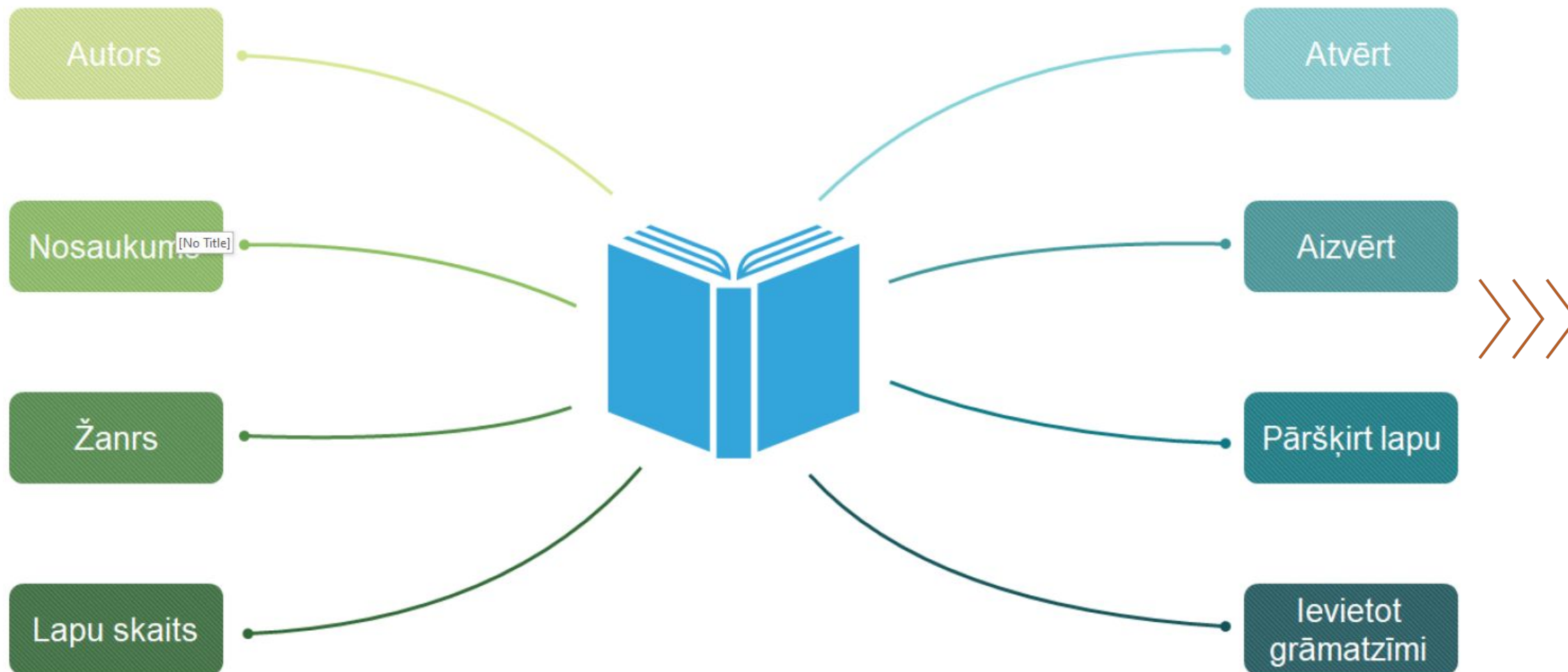
- Objektu būtība
  - Katram objektam ir **īpašības**
  - Objektam var piemist **uzvedība**



# OBJEKTS: SUNS

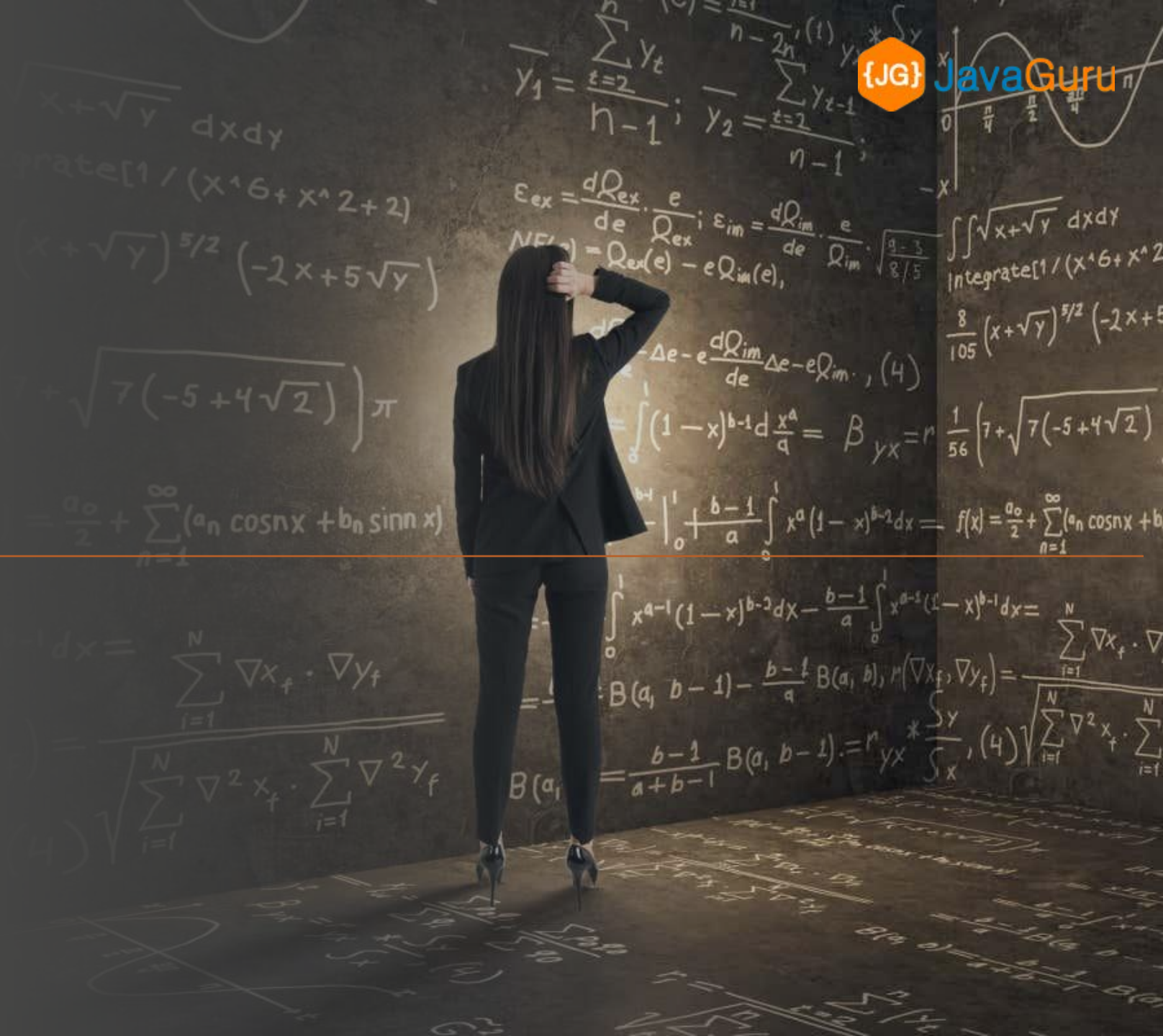


# OBJEKTS: GRĀMATA





# ALGORITMI



# ALGORITMA DEFINĪCIJA

- Algoritms ir procedūra problēmas risināšanai, kuras pamatā ir noteiktu secīgu darbību veikšana
- Matemātikā un informātikā algoritms parasti nozīmē nelielu procedūru, kas atrisina atkārtojošos problēmu
- Datorprogrammu var uzskatīt par izstrādātu algoritmu



# ALGORITHMS: BRAUCIENS TAKSOMETRĀ

1. Sazināties ar taksometru servisu
2. Pateikt atrašanās vietu un galamērķi
3. Gaidīt taksometra ierašanos
4. Iekāpt taksometrā
5. Izbaudīt braucienu
6. Norēķināties par braucienu



# ALGORITMS: IESTĀDĪT KOKU

1. Izrakt bedri
2. Ievietot saknes zemē
3. Apbērt saknes ar zemi
4. Ja zeme ir mitra, tad koka iestādīšana ir pabeigta
5. Apliet koku





# ALOGRITMS: PASŪTĪT ĒDIENU (WOLT)

1. Atrast vēlamo restorānu
2. Izvēlēties un pievienot ēdienu iepirkuma grozam
3. Apstiprināt un apmaksāt pasūtījumu
4. Sagaidīt ēdiena piegādi
5. Izbaudīt maltīti



# JAUTĀJUMI?

---





# PALDIES!

---

