Uniwersytet Wrocławski

Rafał Lewandków

WSTĘP DO PROGRAMOWANIA URZĄDZEŃ MOBILNYCH KOTLIN, JAVA

WYKŁAD 1

- PODSTAWOWE INFORMACJE
- PLAN WYKŁADU
- ZASADY ZALICZENIA

PODSTAWOWE INFORMACJE

Rafał Lewandków
pokój 075
rafal.lewandkow@uwr.edu.pl
rafal.lewandkow2@uwr.edu.pl

Forma zajęć i liczba godzin:
Wykład 15 godz. /Laboratorium 30 godz.

Literatura obowiązkowa i zalecana:

- Java. Efektywne programowanie. Joshua Bloch
- Kotlin w Akcji. Dmitry Jemerov, Svetlana Isakova

Nakład pracy studenta:
Praca własna studenta: 65 godz.
Łączna liczba godzin 125
Liczba punktów ECTS: 5

ZASADY ZALICZENIA

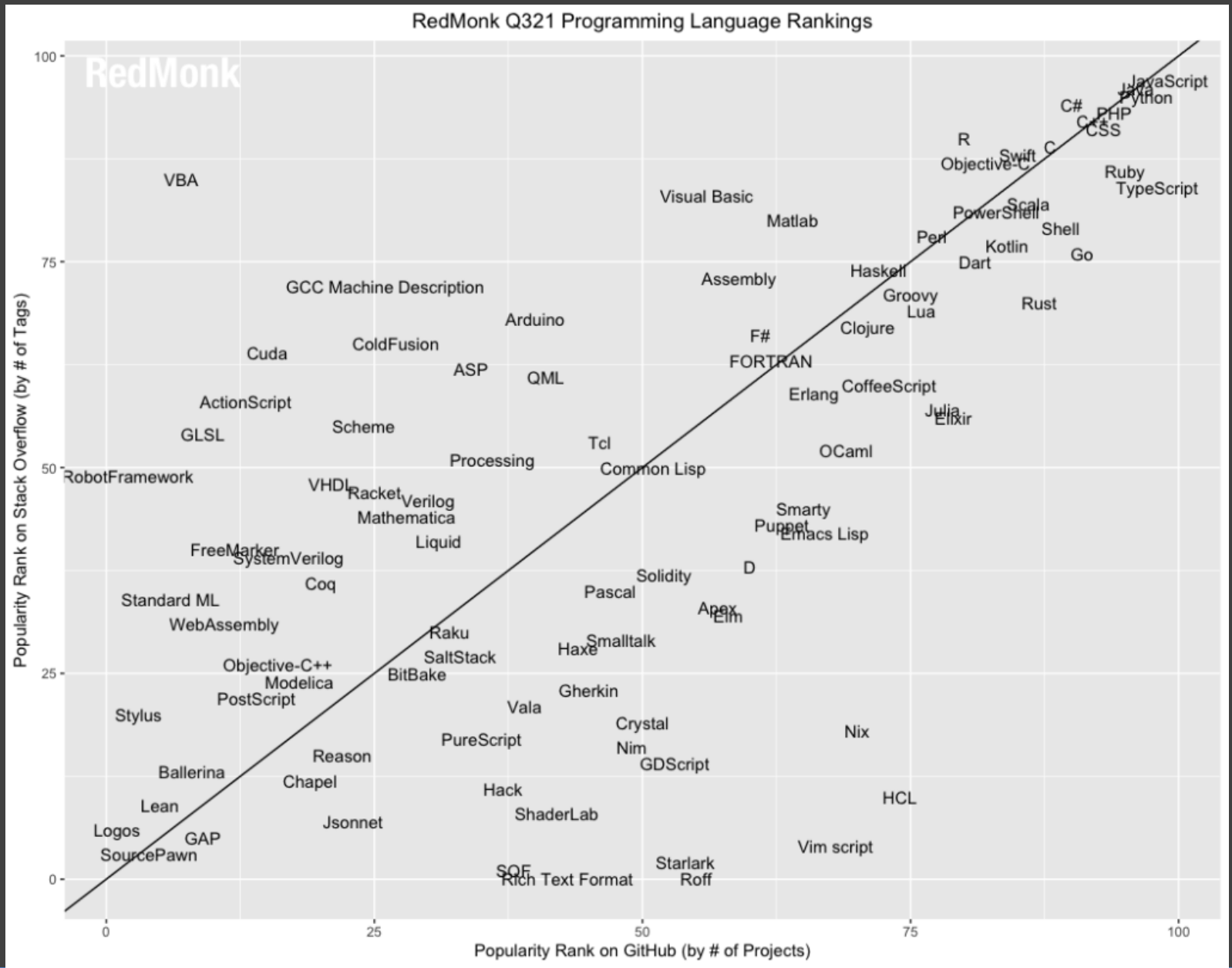
Warunkiem zaliczenia laboratorium jest uzyskanie pozytywnej oceny z wykonywanych zadań umieszczanych na listach

- Na zajęcia przewidzianych jest 8 list zadań
- Z każdej listy wystawiana jest osobna ocena
- Nie jest konieczne zaliczenie wszystkich list aby otrzymać ocenę pozytywną z laboratorium
- Każde zadanie ma przydzieloną liczbę punktów
- Każda lista posiada informację o liczbie punktów wymaganych na konkretną ocenę
- Każda lista posiada termin zwrotu
- Za każdy tydzień opóźnienia otrzymana ocena jest obniżana o 0,5
- Każdą listę można poprawić w ciągu 4 tygodni od terminu zwrotu listy - za wyjątkiem końca semestru
- Poprawa list oddanych po terminie zwrotu jest możliwa pod warunkiem uzyskania co najmniej 50% punktów - termin obowiązuje taki jak w przypadku poprawiania listy oddanej w terminie - ocena nie ulega dalszemu obniżeniu
- Ocena końcowa jest średnią arytmetyczną ze wszystkich ocen z list

- Wyjątkiem jest ocena 3,0 - należy uzyskać średnią co najmniej 3,0

Uniwersytet Wrocławski

Wstęp



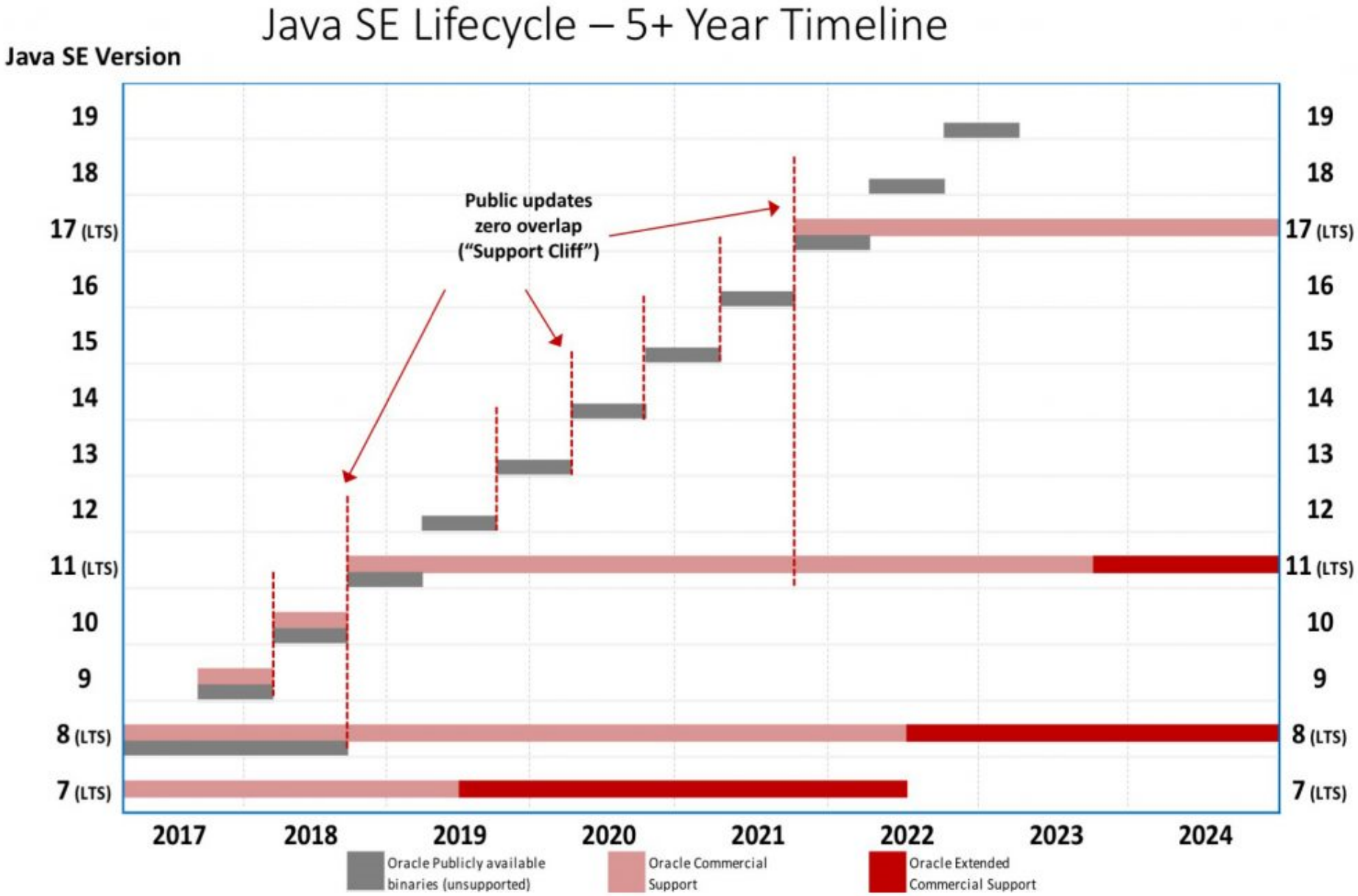
Uniwersytet Wrocławski

Treści Programowe

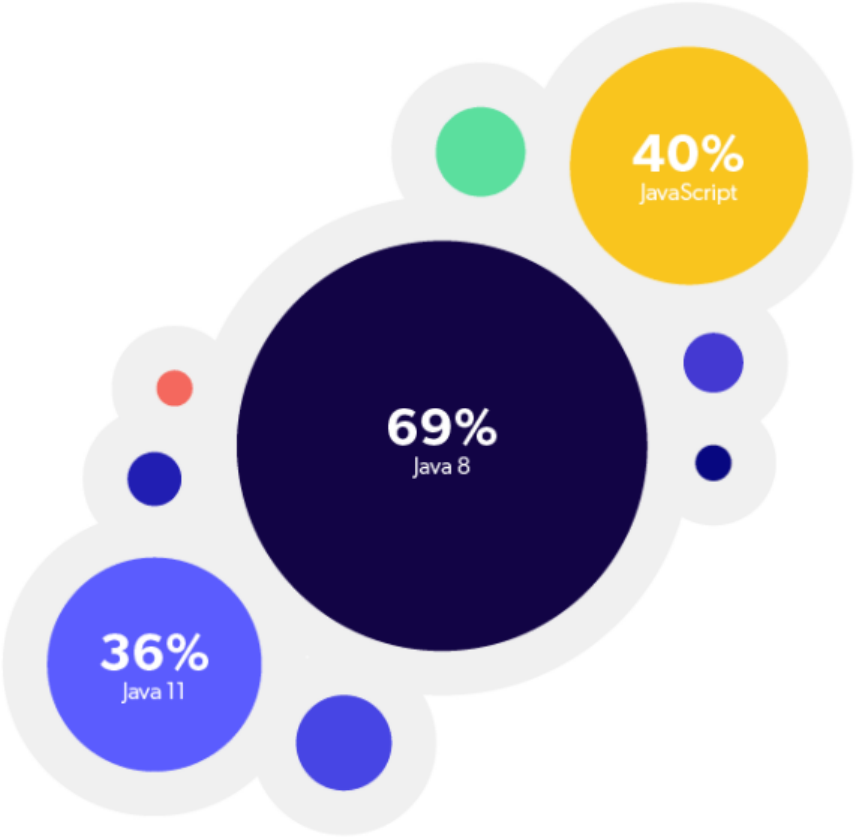
1. Porównanie języków Java i Kotlin
2. Typy danych
3. Wyrażenia, instrukcje, pętle
4. Funkcje
5. Klasy, obiekty
6. Interfejsy
7. Wielowątkowość
8. Wzorce projektowe

IntelliJ Idea

Java


Java SE Lifecycle – 5+ Year Timeline
Java SE Version
19
18
17 (LTS)
16
15
14
13
12
11 (LTS)
10
9
8 (LTS)
7 (LTS)
Public updates
zero overlap
("Support Cliff")
2017
2018
2019
2020
2021
2022
2023
2024
Oracle Publicly available binaries (unsupported)
Oracle Commercial Support
Oracle Extended Commercial Support

Java



Java 8 **69%**

JavaScript **40%**

Java 11 **36%**

Java 12 or Newer **16%**

Java 7 or Older **15%**

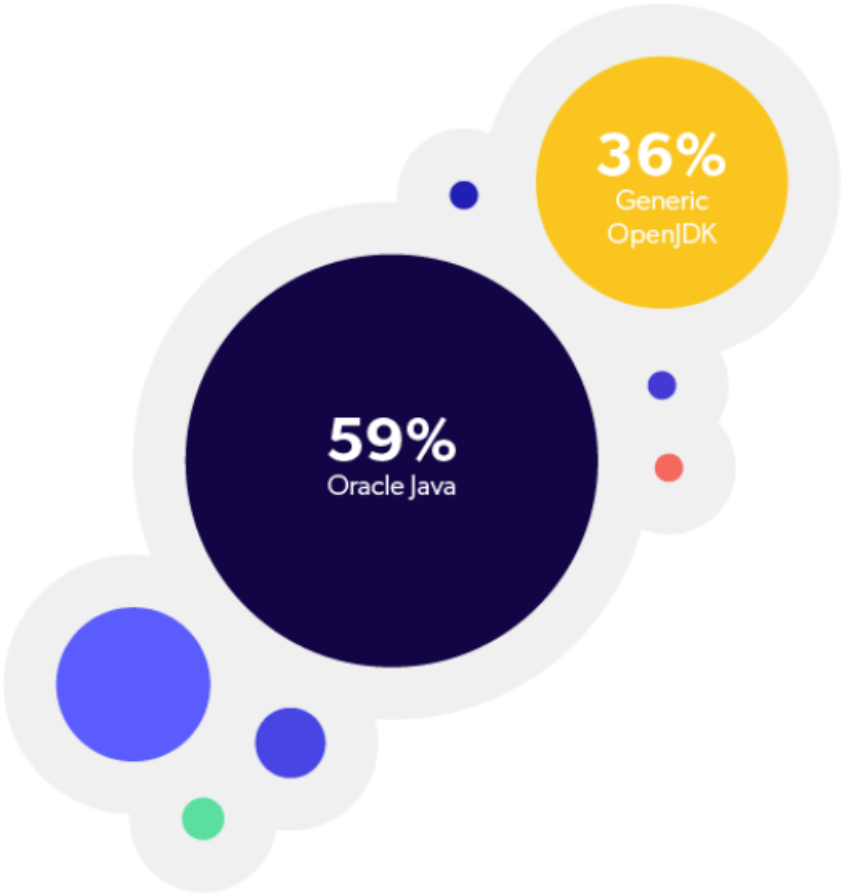
Groovy **10%**

Kotlin **9%**

Scala **6%**

Other **6%**

Java



Oracle Java **59%**

Generic OpenJDK **36%**

Adopt OpenJDK **22%**

Amazon Corretto **10%**

Azul Zulu **6%**

GraalVM **4%**

OpenLogic JDK **4%**

Other **4%**

Java

- Niektóre podobieństwa do C/C++
  - Styl pisania komentarzy
  - Wiele słów kluczowych jest identycznych
  - Typy danych
  - Niemal identyczna struktura kodu
- Niektóre różnice
  - Java wprowadza wiele słów kluczowych nieznanych z C/C++
  - Występują operatory nieznane w C/C++
  - Nie występuje przeciążanie operatorów
  - Brak niektórych cech języka np. wskaźniki
  - Niektóre cechy języka są zmodyfikowane w porównaniu do C/C++ - pętle musza być kontrolowane przez wyrażenia logiczne

Kotlin

- Cechy języka
  - Statyczne typowanie danych – na etapie kompilacji typy wyrażeń są znane a kompilator sprawdza czy istnieją pola i metody w obiektach do których odwołujemy się w kodzie
  - Domniemanie typów – typ danych jest określany przez kompilator na podstawie kontekstu
  - Typy zerowalne
  - Typy funkcyjne
  - Klasy danych
  - Można łączyć programowanie obiektowe i funkcyjne

Kotlin

- Właściwości języka
  - Pragmatyzm – język jest przeznaczony do rozwiązywania praktycznych problemów
  - Jest językiem przemysłowym – nie wprowadza nowych rozwiązań do programowania – wykorzystuje rozwiązania znane z innych języków
  - Nie narzuca stylu/zasad programowania
  - Zwięzłość – im prostszy i krótszy kod tym lepiej
  - Kompatybilność – można wywoływać metody Java
  - Nie posiada własnych bibliotek