

Okruženje za učenje, suradnja
s obitelji i zajednicom u
nastavi informatike /
računalstva u izvarendnim
okolnostima

Alice 3 – primjeri korištenja

dr.sc. Frane Urem prof.v.š.

Oracle Academy Instructor
frane.urem@oracle.com

23-02-2021



Sadržaj

- Uvod
- Alice 3 primjeri korištenja
- Oracle Member Hub
- Što se očekuje u slijedećoj verziji ?
- Pitanja

dr.sc. Frane Urem prof.v.š.

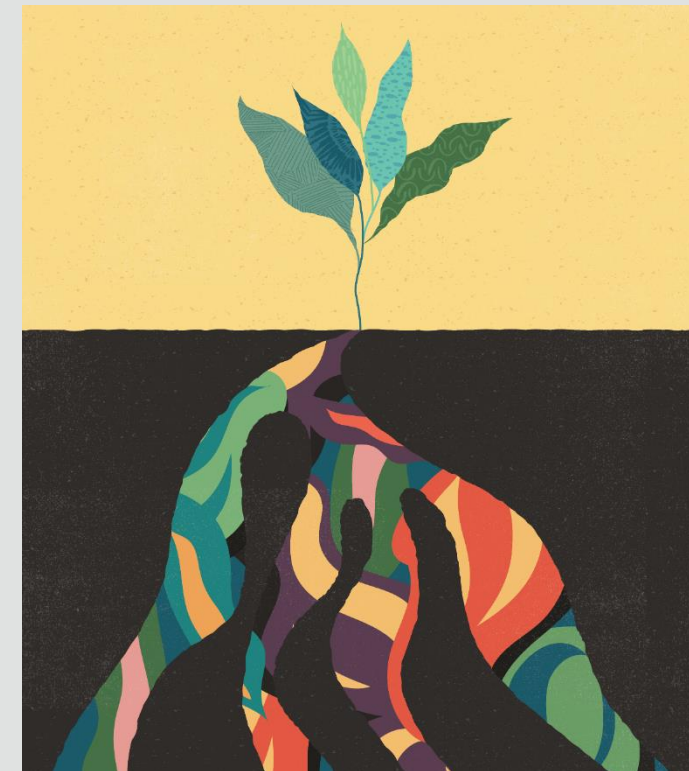
ORACLE
Academy

Veleučilište u Šibeniku – prodekan za poslovanje,
Erasmus+ koordinator, profesor ...

<https://www.linkedin.com/in/frane-urem-09765926/>

Oracle – Oracle Academy trener

<https://academy.oracle.com/en/newsletters/april-2018-full.html#spotlight1>



Uvod

Moderna nastava informatike

- Analiza i usvajanje primjera dobre prakse
- Stručno osposobljavanje nastavnika
- Potpore (projektno financiranje) za razvoj nastave
- Osiguranje interaktivnog nastavnog okruženja
- Dobivanje povratnih informacija o održanoj nastavi od strane učenika
- Osiguranje kvalitete (redovite revizije i unaprjeđenja nastavnih programa)

Samoučenje s potporom instruktora (Supported Self Study)

- U provedbi od ožujka 2021. godine
- Mijenja postojeći model nastavničkih treninga
- Nastavnik putem Member Hub sustava prijavljuje stručno usavršavanje, ali uči i polaže ispite sukladno drugim obvezama
- U svakom trenutku može zatražiti pomoć ili savjet instruktora (Zoom ili e-mail konzultacije)
- Razlog 1: U sklopu postojećeg modela treninga zabilježeno je značajno odustajanje (prosječno 60%) jer nastavnici nisu mogli uskladiti svoje obveze s rasporedom predavanja i polaganja ispita
- Razlog 2: OA instruktori ističu problem nedostatka interakcije sa sudionicima u postojećoj online nastavi

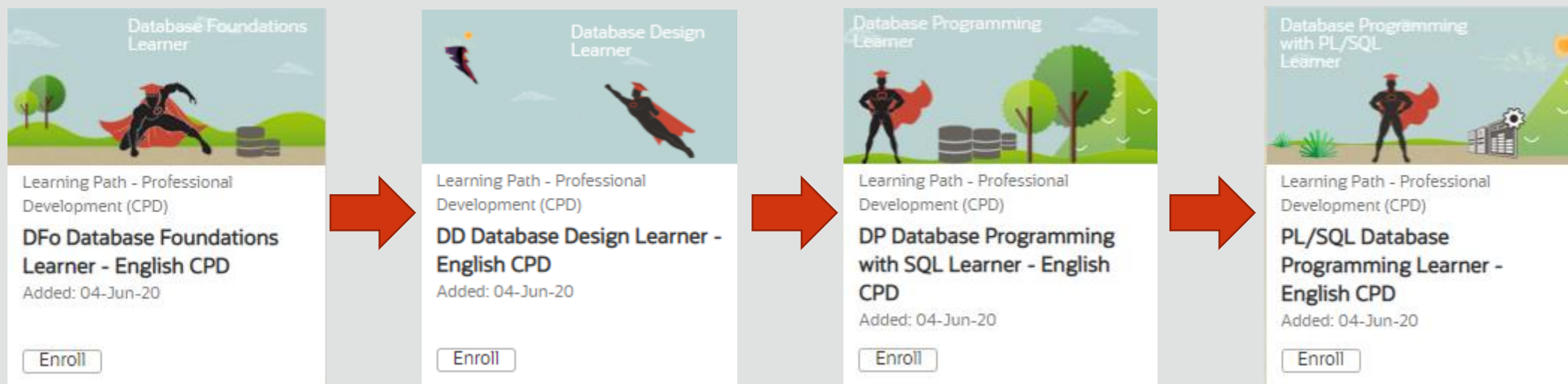
Samoučenje s potporom instruktora (Supported Self Study)

- Programi nastavničkog usavršavanja tzv. Learning Path - Professional Development (CPD)
- Java



Samoučenje s potporom instruktora (Supported Self Study)

- Programi nastavničkog usavršavanja tzv. Learning Path - Professional Development (CPD)
- Baze podataka



Samoučenje s potporom instruktora (Supported Self Study)

- Programi nastavničkog usavršavanja tzv. Learning Path - Professional Development (CPD)
- Razvoj web i mobilnih aplikacija (APEX), strojno učenje (engl. machine learning), vođenje projekata (Primavera)



Application Development Foundations Learner

Learning Path - Professional Development (CPD)

ADFo Application Development Foundations Learner - English CPD

Added: 04-Jun-20

[Enroll](#)




Artificial Intelligence with Machine Learning in Java Learner

Learning Path - Professional Development (CPD)

AiML Artificial Intelligence with Machine Learning in Java Learner - English CPD

Added: 27-Apr-20

[Enroll](#)



Oracle Primavera Learner

Learning Path - Professional Development (CPD)

Primavera P6 Professional Fundamentals - English CPD

Added: 04-Jun-20

[Enroll](#)

Samoučenje s potporom instruktora (Supported Self Study)

- Programi nastavničkog usavršavanja tzv. Learning Path - Professional Development (CPD)
- Projektne radionice (izvanškolske aktivnosti učenika)



Workshops in a Box -
Finch Greenfoot

Learning Path - Professional
Development (CPD)

**WIB Workshop -
Programming the Finch Robot
in Greenfoot - English CPD**

Added: 08-Jun-20

Enroll



Workshops in a Box -
Solve it with SQL

Learning Path - Professional
Development (CPD)

**WIB Workshop - Solve It with
SQL - English CPD**

Added: 04-Jun-20

Enroll



Workshops in a Box -
Finch Java

Learning Path - Professional
Development (CPD)

**WIB Workshop -
Programming the Finch Robot
in Java - English CPD**

Added: 08-Jun-20

Enroll



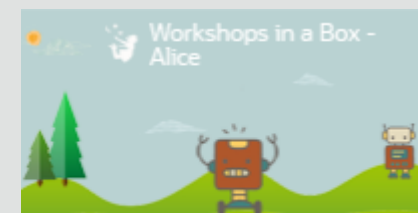
Workshops in a Box -
Greenfoot

Learning Path - Professional
Development (CPD)

**WIB Workshop - Creating Java
Programs with Greenfoot -
English CPD**

Added: 04-Jun-20

Enroll



Workshops in a Box -
Alice

Learning Path - Professional
Development (CPD)

**WIB Workshop - Getting
Started with Java Using Alice -
English CPD**

Added: 04-Jun-20

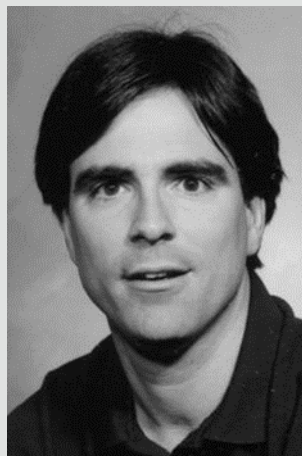
Enroll

Alice 3 primjeri korištenja

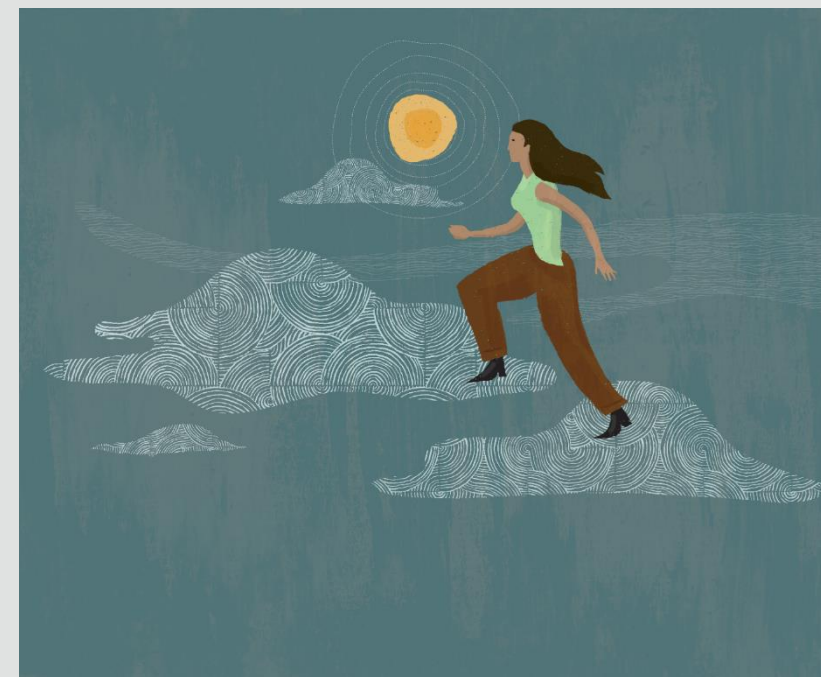


Alice 3

Idejni tvorac Alice platforme:
prof. Randy Pausch - Sveučilište Carnegie Mellon
(1960. – 2007.)



The Last Lecture:
Really Achieving Your Childhood
Dreams

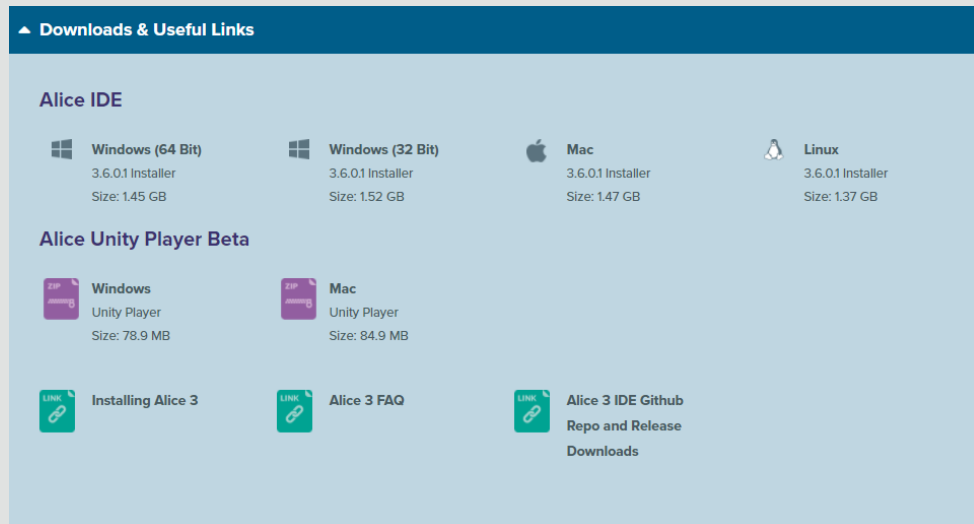
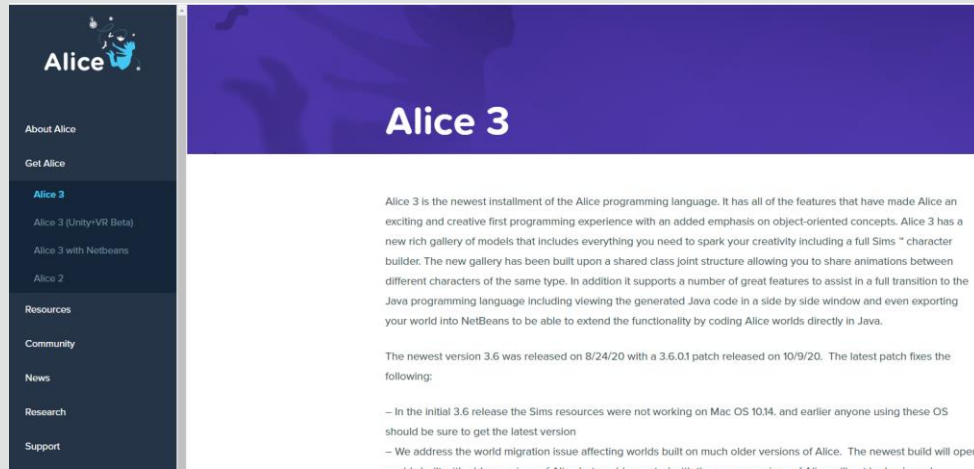


Dobra iskustva s Alice 3 platformom

- Jednostavna instalacija i početak rada
- Pristup nastavnim resursima i globalnim primjerima dobre prakse
- Inovativno i interaktivno nastavno okruženje – utječe na motiviranost učenika
- Projektna nastava



Alice 3



- Instalacija

<https://www.alice.org/get-alice/alice-3>

- Uloga programera je nalik ulozi redatelja filma ili video-igre

- Jednostavno usvajanje osnovnih pojmova (klasa, objekt, metode, svojstva ...)

Zašto Alice 3 (animacija) ?

Korak 1:
Definicija
scenarija

Korak 2:
Razvoj
scenarija

Korak 3:
Programiranje
animacije

Korak 4:
Pokretanje
animacije

Korak 1:
Uočavanje
problema

Korak 2:
Razvoj
rješenja

Korak 3:
Provedba
rješenja

Korak 4:
Provjera i
ocjena rješenja

- Planiranje i razvoj kompliciranih animacija je vrlo složen proces i usporediv s razvojem stvarnih aplikacija
- Kako bi se takav posao pojednostavnio potrebno je analizirati postavljeni zadatak i uočiti sve probleme koji se moraju savladati

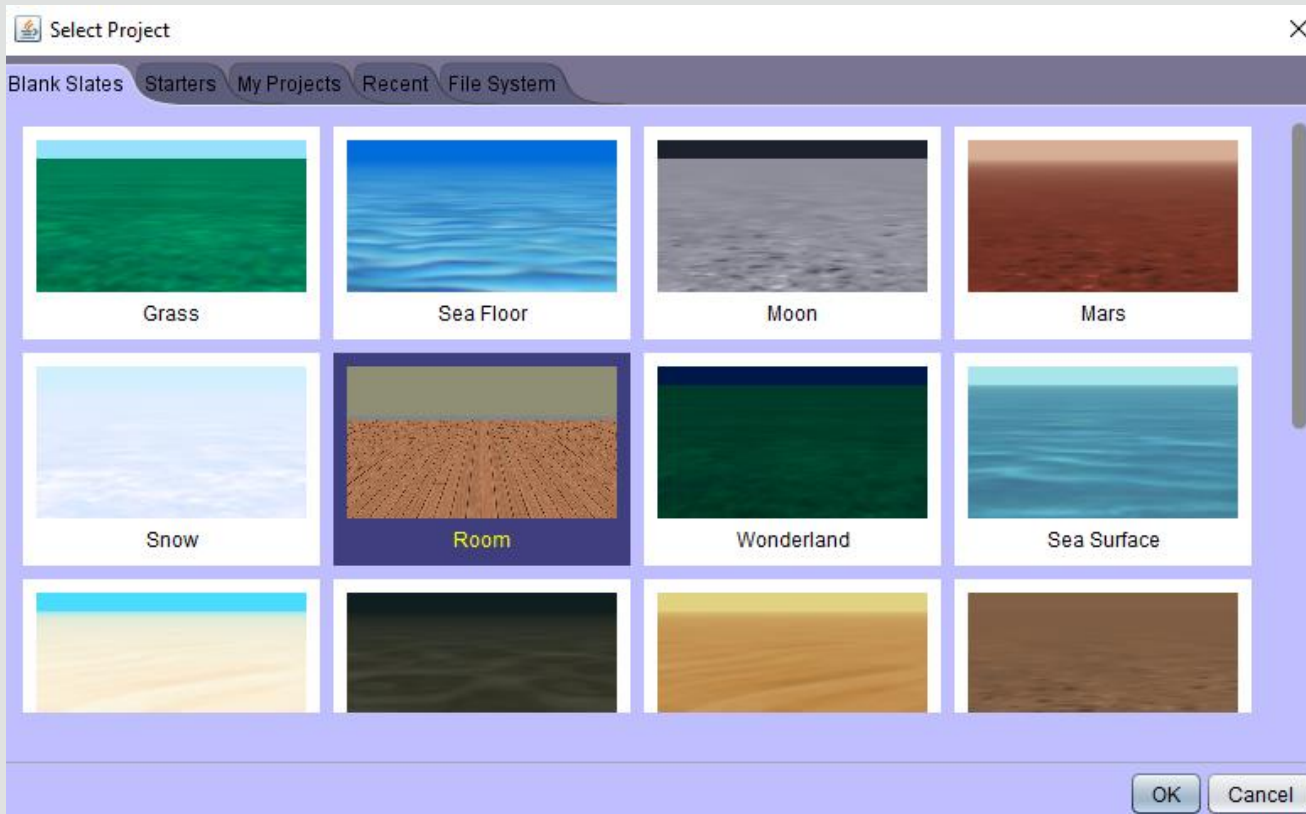
Definicija i razvoj scenarija

- U razvoju softvera je posebno bitan proces razvoja scenarija - ili priče - koja mu daje svrhu, npr.
 - Priča koja predstavlja problem i rješenje
 - Pouka za podučavanje pojma matematike
 - Postupak za simulaciju ili demonstraciju
 - Igra za zabavu ili trening

Definicija i razvoj scenarija

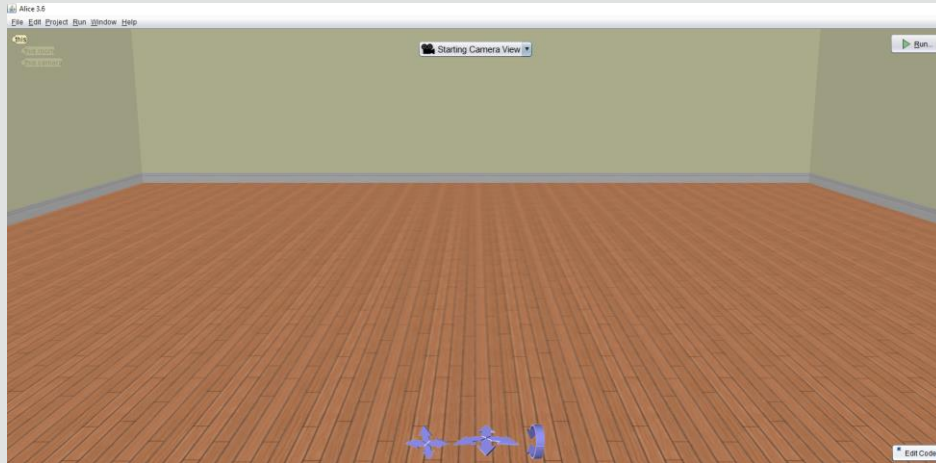
- Identificiraju se specifikacije dizajna za scenarij animacije, npr. kako se objekti pojavljuju, kreću, govore, komuniciraju itd.
- Za planiranje Alice 3 animacije koriste se dvije vrste scenarija :
 - Vizualni: niz ilustriranih slika koje predstavljaju glavne prizore animacije
 - Tekstualni: Detaljan, naručeni popis radnji koje svaki objekt obavlja u svakoj sceni animacije

Otvaranje novog projekta

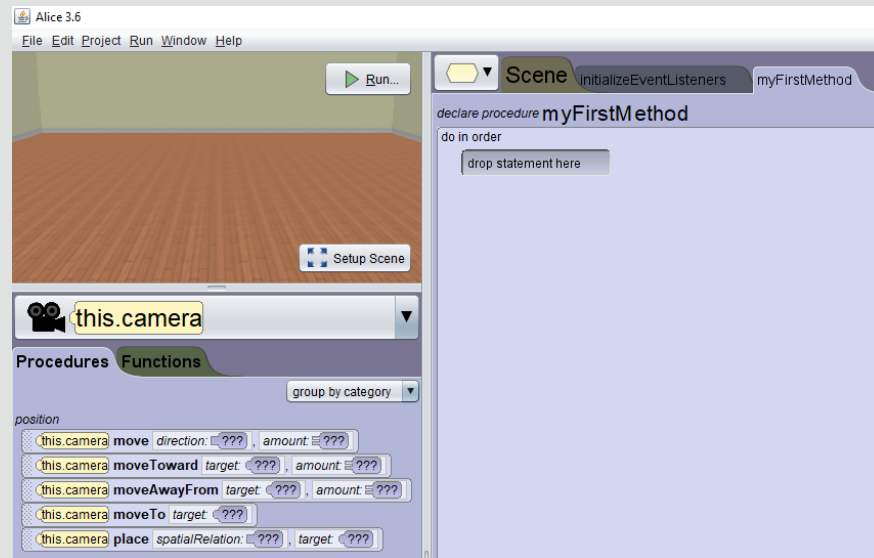


- Pokrenite Alice 3
- U početnom dijalogu (Welcome dialog box) odaberite karticu Blank States
- Odaberite željeni predložak, i potvrdite s OK

Osnovni dijelovi razvojne okoline



- Prijelaz između različitih pogleda je moguć odabirom gumba Edit Code (iz uređivača scene) ili Setup Scene (iz uređivača programskog koda)



5. razred osnovne škole

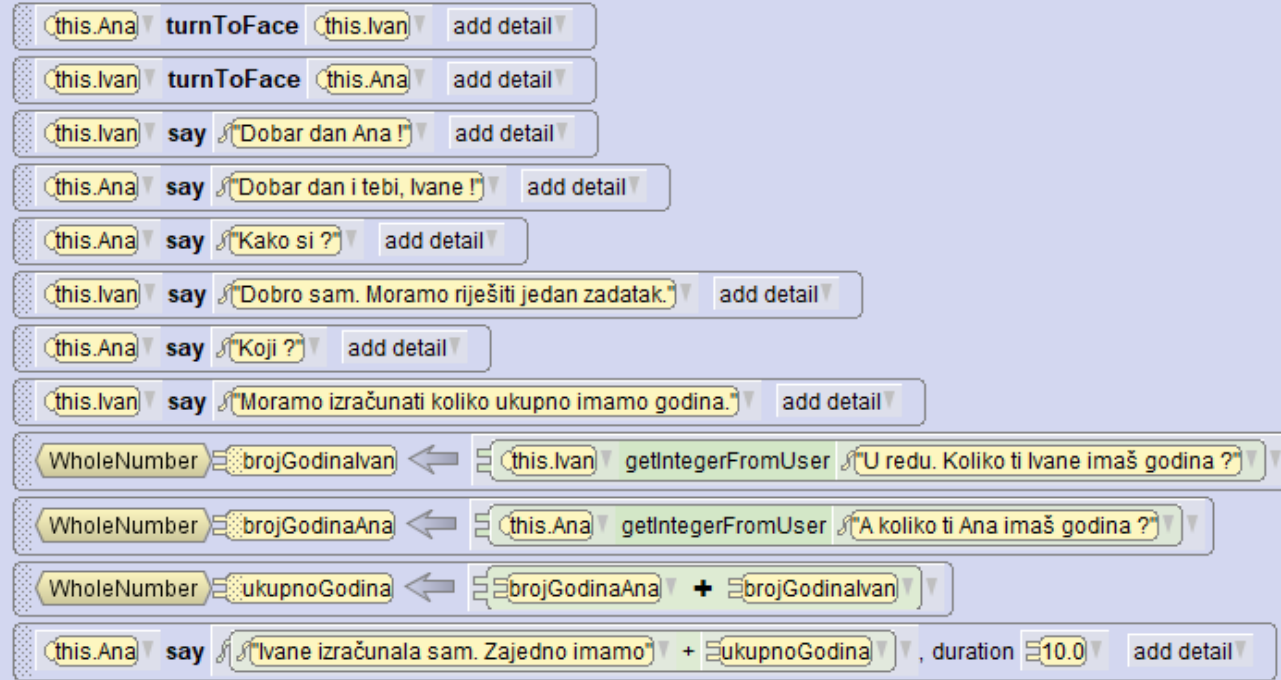
- **Računalno razmišljanje i programiranje**
- *B.5.1. Nakon pete godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik se koristi programskim alatom za stvaranje programa u kojemu se koristi ulaznim i izlaznim vrijednostima te ponavljanjem*
- *Samostalno ili uz pomoć učitelja rješavati jednostavne probleme koji se koriste ulaznim i izlaznim vrijednostima, npr. računanje s ulaznim vrijednostima, stvaranje dijaloga među likovima (objektima).*
- **Primjer:** Ana i Ivan razgovaraju u sobi. Ana pita Ivana koliko ima godina. Ivan pita Anu koliko ima godina. Ana izračuna i kaže koliko zajedno imaju godina.

5. razred osnovne škole

- Učenici rade u paru
- Oblikuju i postavljaju likove na scenu.
- Dogovaraju osnovni scenarij – što koji lik mora reći i kojim redoslijedom.
- Definiraju varijable u kojima se bilježe godine starosti za oba lika
- Postavljaju naredbe za unos podataka i računanje.
- Realiziraju prikaz rezultata

5. razred osnovne škole

do in order



6. razred osnovne škole

- **Računalno razmišljanje i programiranje**
- *B.6.1. Nakon šeste godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik stvara, prati i preuređuje programe koji sadrže strukture grananja i uvjetnoga ponavljanja te predviđa ponašanje jednostavnih algoritama koji mogu biti prikazani dijagramom, riječima govornoga jezika ili programskim jezikom.*
- *Učenik samostalno ili uz pomoć učitelja analizira zadani problem te predlaže neko algoritamsko rješenje, rješenje problema prikazuje dijagramom, riječima govornoga jezika ili naredbama programskoga jezika.*
- **Primjer:** Ana pita učenika da joj kaže tri broja. Nakon toga prikazuje koji je od ta tri broja najmanji.

6. razred osnovne škole

- Učenik samostalno stvara program/algoritam kao rješenje problema koje uključuje niz uputa (naredbi) primjenom svih algoritamskih struktura
- Predviđa odgovarajuće ulazne (testne) primjer
- Kritički provjerava ispravnost rješenja i prema potrebi preuređuje svoje rješenje.



6. razred osnovne škole



Run

speed: 1x

Molim Vas unesite drugi broj

2

7 8 9
4 5 6
1 2 3
0 ±

OK

Molim Vas unesite prvi broj

7

7 8 9
4 5 6
1 2 3
0 ±

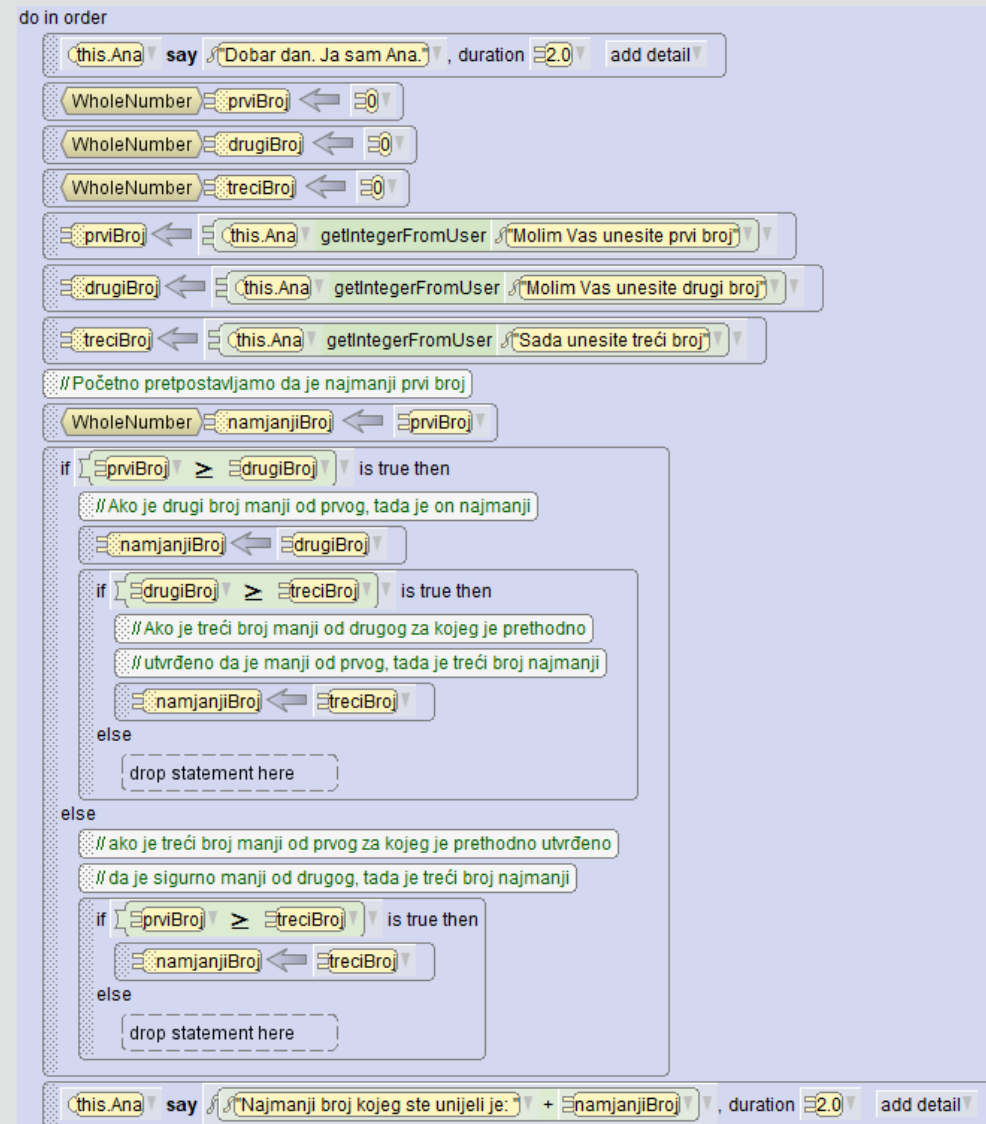
OK

Sada unesite treći broj

9

7 8 9
4 5 6
1 2 3
0 ±

OK



7. razred osnovne škole

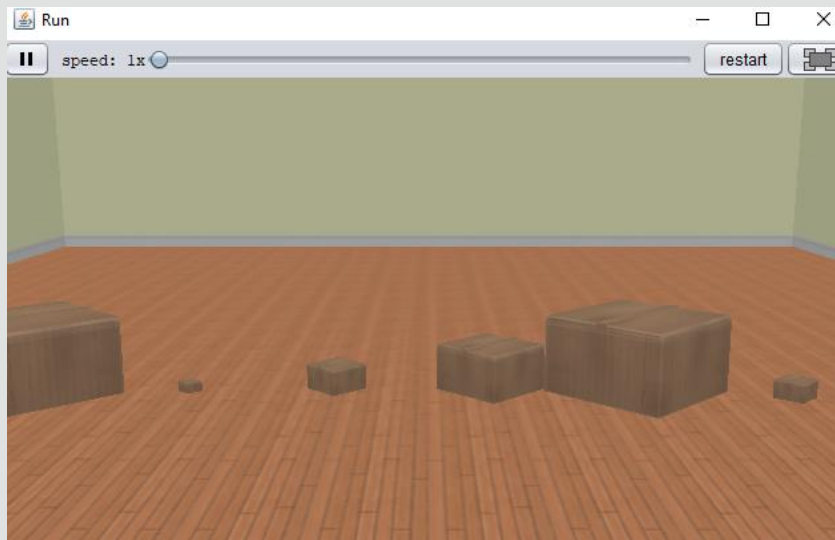
- **Računalno razmišljanje i programiranje**
- *B.7.2 Nakon sedme godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik primjenjuje algoritam (sekvencijalnoga) pretraživanja pri rješavanju problema.*
- *Prepoznati i opisati kriterij pretraživanja za zadani problem. Analizirati potrebne algoritamske strukture za rješavanje problema. Pokazati postupak pretraživanja koristeći se alatima vizualizacije.*
- **Primjer:** Postaviti šest objekata na scenu (npr. obične kutije). U prvom pretraživanju postaviti da svi objekti budu jednake visine. U drugom pretraživanju postaviti slučajne visine objekata.



7. razred osnovne škole

- Učenik pronalazi/predviđa slične probleme na koje se odabrani algoritam (sekvencijalnoga) pretraživanja može primijeniti
- Utvrđuje važna obilježja toga algoritma pretraživanja
- Učenik vizualizira rezultate pretraživanja

7. razred osnovne škole



```
do in order
  // postavljanje brojača
  WholeNumber brojac ← 0
  // Stvaranje kolekcije objekata
  Box[] listaKutija ← new Box[] { this.box, this.box2, this.box3, this.box4, this.box5, this.box6 }
  // Postavljanje jednake visine za sve kutije
  while brojac ≤ 5 is true
    listaKutija [ brojac ] setHeight 0.5 add detail
    brojac ← brojac + 1
  loop
  // Postavljanje slučajnih visina
  brojac ← 0
  while brojac ≤ 5 is true
    listaKutija [ brojac ] setHeight nextRandomRealNumberInRange 0.0, 0.5 add detail
    brojac ← brojac + 1
  loop
```

8. razred osnovne škole

- **Računalno razmišljanje i programiranje**
- *B.8.3 Nakon osme godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik prepoznaje i opisuje mogućnost primjene rekurzivnih postupaka pri rješavanju odabranih problema te istražuje daljnje mogućnosti primjene rekurzije.*
- *Učenik promatra i opisuje zajednička obilježja nekih rekurzivnih fenomena te poznaje korake rekurzivnoga postupka. Analizira odabrani problem te u njemu identificira raosnovi slučaj rekurzije te način rekurzivnoga pozivanja.*
- **Primjer:** Postaviti dva objekta na scenu (zec i štap). Realizirati pomicanje zeca prema štapu primjenom rekurzivnog postupka u koracima od 0,25 m. Zec se zaustavlja kada je blizu štapa na udaljenosti od 0,20 m.

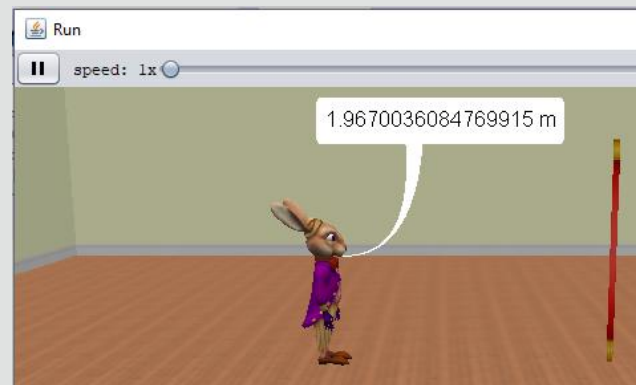
8. razred osnovne škole

- Učenik pronalazi i predlaže (grafički ili riječima/uputama) rješenje odabranoga problema primjenom rekurzivnoga postupka

8. razred osnovne škole

```
declare procedure myFirstMethod
do in order
  // glavni program
  this.zec pointAt this.stap add detail
  this provjeriUdaljenost
```

```
declare procedure provjeriUdaljenost Add Parameter...
do in order
  // rekurzivna funkcija za pomicanje zeca
  DecimalNumber udaljenost ← this.zec getDistanceTo this.stap
  this.zec say udaljenost + "m" add detail
  // provjera udaljenost između objekata. ako je manja od 0,25 m pomicanje se nastavlja
  if udaljenost > 0.25 is true then
    this.zec move FORWARD, 0.25 add detail
    // procedura poziva samu sebe - rekurzija !
    this provjeriUdaljenost
  else
    // zec je stigao na odredište, pomicanje se zaustavlja
    this.zec say Stigao sam. Dosta je rekurzije !
```



Srednje škole - prva godina učenja

- **Računalno razmišljanje i programiranje**
- *B.1.2 Nakon prve godine učenja predmeta Informatika u srednjoj školi u domeni Računalno razmišljanje i programiranje učenik primjenjuje jednostavne tipove podataka te argumentira njihov odabir, primjenjuje različite vrste izraza, operacija, relacija i standardnih funkcija za modeliranje jednostavnoga problema u odabranome programskom jeziku.*
- *Primjenjuje standardne operacije i funkcije nad jednostavnim tipovima podataka pri rješavanju zadanoga problema te funkcije unosa i ispisa.*
- **Primjer:** Inicijalizirati objekt koji predstavlja krug. Upravljači veličinom kruga preko unosa radijusa. Realizirati posebne funkcije za računanje opsega i površine kruga.

Srednje škole - prva godina učenja

- Modelira jednostavnije matematičke i logičke probleme odgovarajućim izrazima.
- Argumentira prednosti korištenja odabranoga tipa podataka s obzirom na druge tipove podataka.



Srednje škole - prva godina učenja

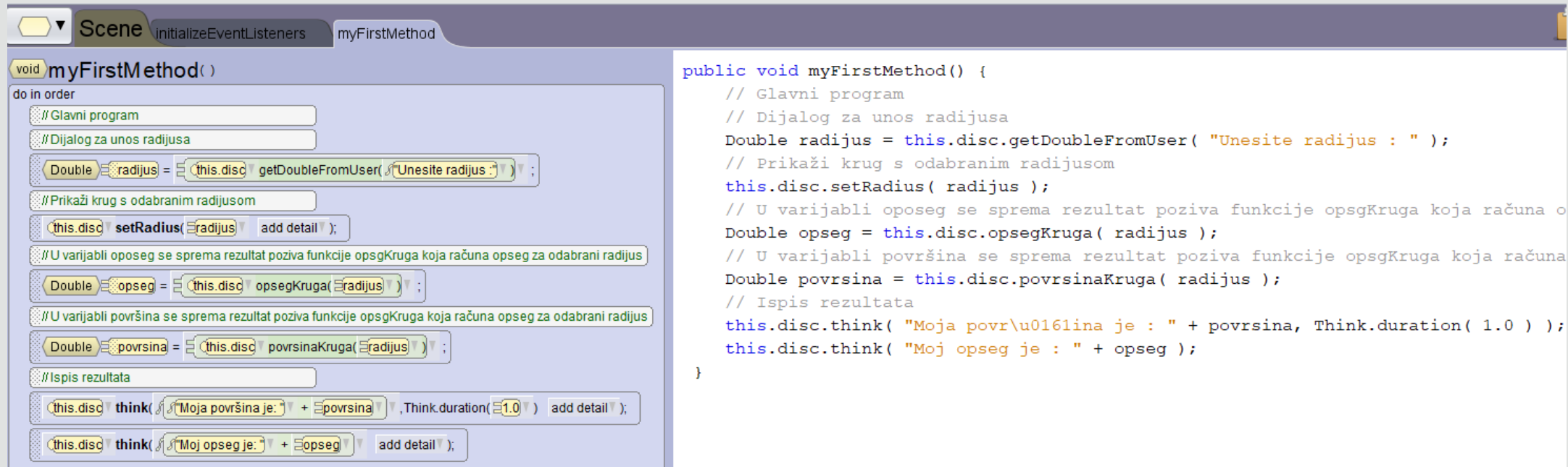
```
do in order
// Glavni program
// Dijalog za unos radijusa
DecimalNumber radius ← this.disc.getDoubleFromUser "Unesite radijus:"
// Prikaži krug s odabranim radijusom
this.disc.setRadius radius add detail
// U varijabli opseg se sprema rezultat poziva funkcije opsegKrug koja računa opseg za odabrani radijus
DecimalNumber opseg ← this.disc.opsegKrug radiusKrug: radius
// U varijabli površina se sprema rezultat poziva funkcije opsegKrug koja računa opseg za odabrani radijus
DecimalNumber površina ← this.disc.povrsinaKrug radiusKrug: radius
// Ispis rezultata
this.disc.think "Moja površina je: " + površina add detail
this.disc.think "Moj opseg je: " + opseg add detail
```

```
declare DecimalNumber function opsegKrug with parameter: DecimalNumber radiusKrug
do in order
// računanje opsega kruga za zadani radijus
return new DecimalNumber { PI * 2.0 * radiusKrug }
```

```
DecimalNumber povrsinaKrug( DecimalNumber radiusKrug
do in order
// računanje površine kruga za zadani radijus
return PI * radiusKrug * radiusKrug
```



Srednje škole - prva godina učenja – odabrani programski jezik Java



Scene initializeEventListeners myFirstMethod

void myFirstMethod()

do in order

- // Glavni program
- // Dijalog za unos radijusa
- Double radius = this.disc.getDoubleFromUser("Unesite radijus : ");
- // Prikaži krug s odabranim radijusom
- this.disc.setRadius(radius);
- // U varijabli opseg se sprema rezultat poziva funkcije opsegKrug koja računa opseg za odabrani radijus
- Double opseg = this.disc.opsegKrug(radius);
- // U varijabli površina se sprema rezultat poziva funkcije opsegKrug koja računa opseg za odabrani radijus
- Double površina = this.disc.povrsinaKrug(radius);
- // Ispis rezultata
- this.disc.think("Moja površina je: " + površina, Think.duration(1.0));
- this.disc.think("Moj opseg je: " + opseg);

```
public void myFirstMethod() {  
    // Glavni program  
    // Dijalog za unos radijusa  
    Double radius = this.disc.getDoubleFromUser( "Unesite radijus : " );  
    // Prikaži krug s odabranim radijusom  
    this.disc.setRadius( radius );  
    // U varijabli opseg se sprema rezultat poziva funkcije opsegKrug koja računa o  
    Double opseg = this.disc.opsegKrug( radius );  
    // U varijabli površina se sprema rezultat poziva funkcije opsegKrug koja računa  
    Double površina = this.disc.povrsinaKrug( radius );  
    // Ispis rezultata  
    this.disc.think( "Moja površina je: " + površina, Think.duration( 1.0 ) );  
    this.disc.think( "Moj opseg je: " + opseg );  
}
```



Srednje škole - druga godina učenja

- **Računalno razmišljanje i programiranje**
- *B.2.2 Nakon druge godine učenja predmeta Informatika u srednjoj školi u domeni Računalno razmišljanje i programiranje učenik analizira osnovne algoritme s jednostavnim tipovima podataka i osnovnim programskim strukturama i primjenjuje ih pri rješavanju novih problema.*
- *Samostalno rješavanje zadataka s primjenom standardnih algoritama: zbrajanje/množenje prirodnih brojeva unutar određenoga intervala, unos i zbrajanje/množenje n brojeva, provjera je li broj prost, najveća unesena vrijednost, prebrojavanje po zadanom kriteriju, odvajanje znamenaka prirodnog broja.*
- **Primjer:** Omogućiti unos n cijelih brojeva. Prebrojati koliko ima parnih, a koliko neparnih brojeva. Odrediti najmanji i najveći broj.

Srednje škole - druga godina učenja

do in order

```
Integer brojParnih = 0 ;
Integer brojNeparnih = 0 ;
Integer najmanjiBroj = 0 ;
Integer[] niz = new Integer[] {0,0,0,0,0,0,0,0,0,0} ;
Integer najvećiBroj = 0 ;

for (Integer indexA = 0; indexA < 10; indexA++) {
    niz [ indexA ] = this.box.getIntegerFromUser( "Unesite element niza : " + indexA + " : " + niz [ indexA ] );
}

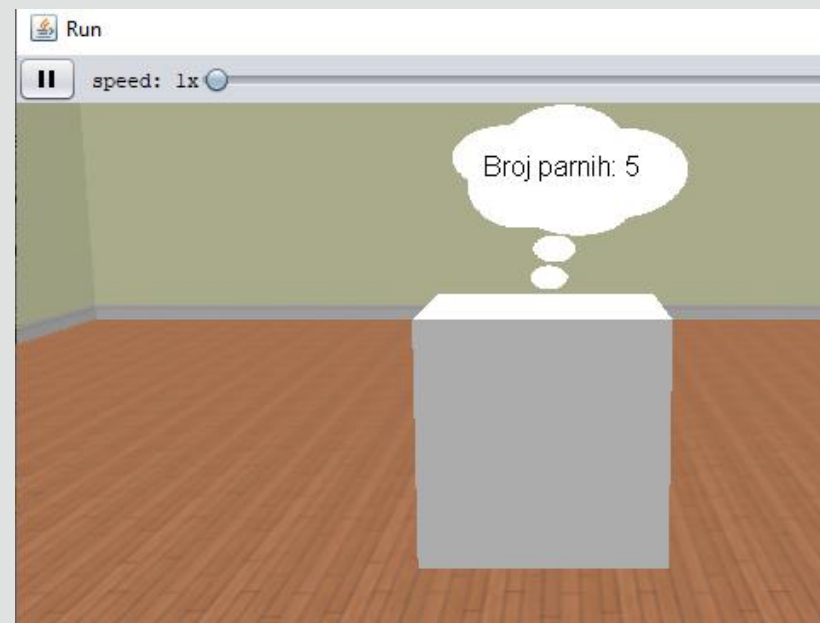
najvećiBroj = niz [ 0 ] ;
najmanjiBroj = niz [ 0 ] ;

for (Integer indexB = 0; indexB < 10; indexB++) {
    if ( niz [ indexB ] % 2 == 0 ) {
        brojParnih = brojParnih + 1 ;
    } else {
        brojNeparnih = brojNeparnih + 1 ;
    }

    if ( niz [ indexB ] < najmanjiBroj ) {
        najmanjiBroj = niz [ indexB ] ;
    } else {
        drop statement here
    }

    if ( niz [ indexB ] > najvećiBroj ) {
        najvećiBroj = niz [ indexB ] ;
    } else {
        drop statement here
    }
}
```

```
this.box.think( "Broj neparnih: " + brojNeparnih ); add detail ;
this.box.think( "Broj parnih: " + brojParnih ); add detail ;
this.box.think( "Najveći broj: " + najvećiBroj ); add detail ;
this.box.think( "Najmanji broj: " + najmanjiBroj ); add detail ;
```



Srednje škole - treća godina učenja

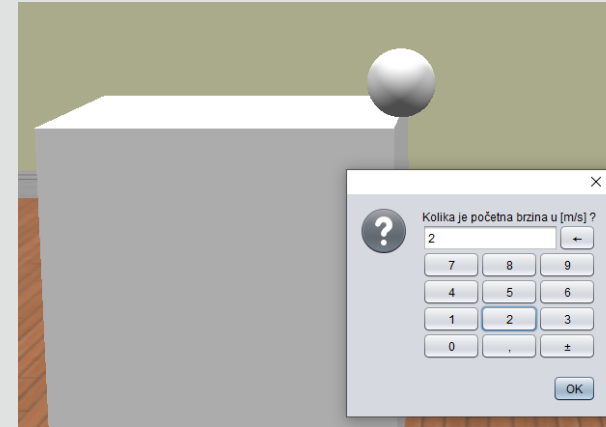
- **Računalno razmišljanje i programiranje**
- *B.3.3. Nakon treće godine učenja predmeta Informatika u srednjoj školi u domeni Računalno razmišljanje i programiranje učenik koristeći neki grafički modul vizualizira i grafički prikazuje neki problem iz svoje okoline*
- *Crtanje kompozicija različitih složenosti. Eksperimentiranje s mogućnostima prikaza boje. Kako postići animaciju? Animirati kretanje objekta kojom zamišljenom krivuljom, npr. kretanje brodića sinusoidalnom putanjom. Crtanje matematičkih funkcija u različitim koordinatnim sustavima. Realizacija koje jednostavne igre poput potrage za blagom ili križić-kružića.*
- **Primjer:** Simulirati horizontalni hitac kugle koja pada se visine od jednog metra. Omogućiti unos početne brzine.

Srednje škole - treća godina učenja

do in order

```
Double pocetnaBrzina = 1.0 ;  
Double y0 = 1.0 ;  
Double y = 0.0 ;  
Double x = 0.0 ;  
Double t = 0.0 ;  
final Double g = 9.81 ;  
pocetnaBrzina = this.box.getDoubleFromUser( "Kolika je početna brzina u [m/s] ?" ) ;
```

```
while( y < y0 ){  
    x = pocetnaBrzina * t ;  
    y = 0.5 * g * t * t ;  
    //istovremeno gibanje po osima x i y  
    ThreadUtilities.doTogether( ()-> {  
        this.sphere.move( MoveDirection.LEFT, x, addDetail );  
    }, 0-> {  
        this.sphere.move( MoveDirection.DOWN, y, addDetail );  
    } );  
    //isprekidano gibanje po osima x i y  
    /*do in order*/ {  
        this.sphere.move( MoveDirection.LEFT, x, addDetail );  
        this.sphere.move( MoveDirection.DOWN, y, addDetail );  
    }  
    t = t + 0.1 ;  
}
```

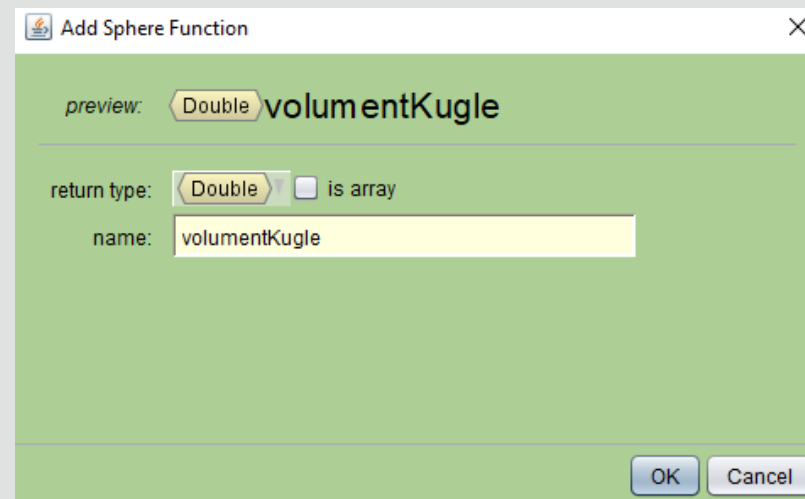
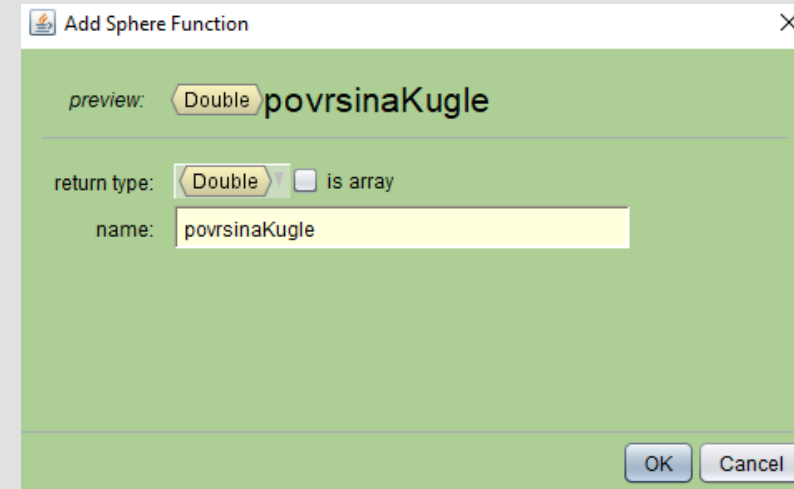
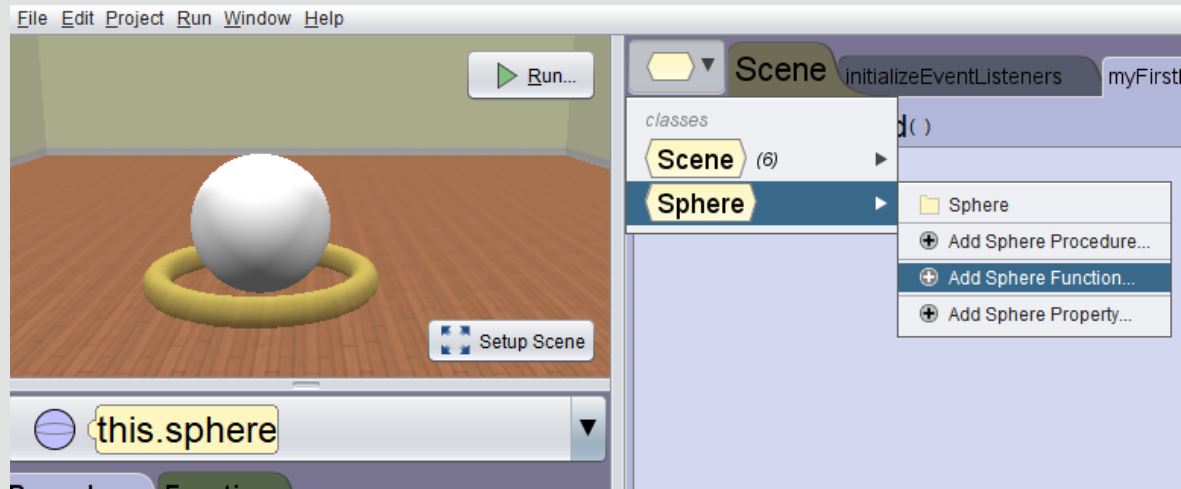


```
this.box.think( "Vrijeme pada: " + t, addDetail );  
this.box.think( "Domet: " + x, addDetail );
```

Srednje škole - četvrta godina učenja

- **Računalno razmišljanje i programiranje**
- *B.4.3. Nakon četvrte godine učenja predmeta Informatika u srednjoj školi u domeni Računalno razmišljanje i programiranje učenik osmišljava objektni model s pripadnim složenim strukturama podataka implementira ga u zadanome programskom jeziku.*
- *Objektno usmjereno programiranje moguće je uvesti korištenjem nekih od alata poput alata Alice 3. Stvarati jednostavne klase s pripadnim metodama i svojstvima za neke standardne metode (pravokutnik, trokut, razlomak, kompleksni broj i sl.). Stvarati kompleksni model za složenije probleme. Primijeniti objektno programiranje u grafici.*
- **Primjer:** Proširiti postojeću Alice 3 klasu Sphere s metodama za računanje volumena (*volumenKugle*) i površine (*povrsinaKugle*).

Srednje škole - četvrta godina učenja



Za one koji žele znati malo više

Oracle Member Hub

Alice 3 nastavni materijali

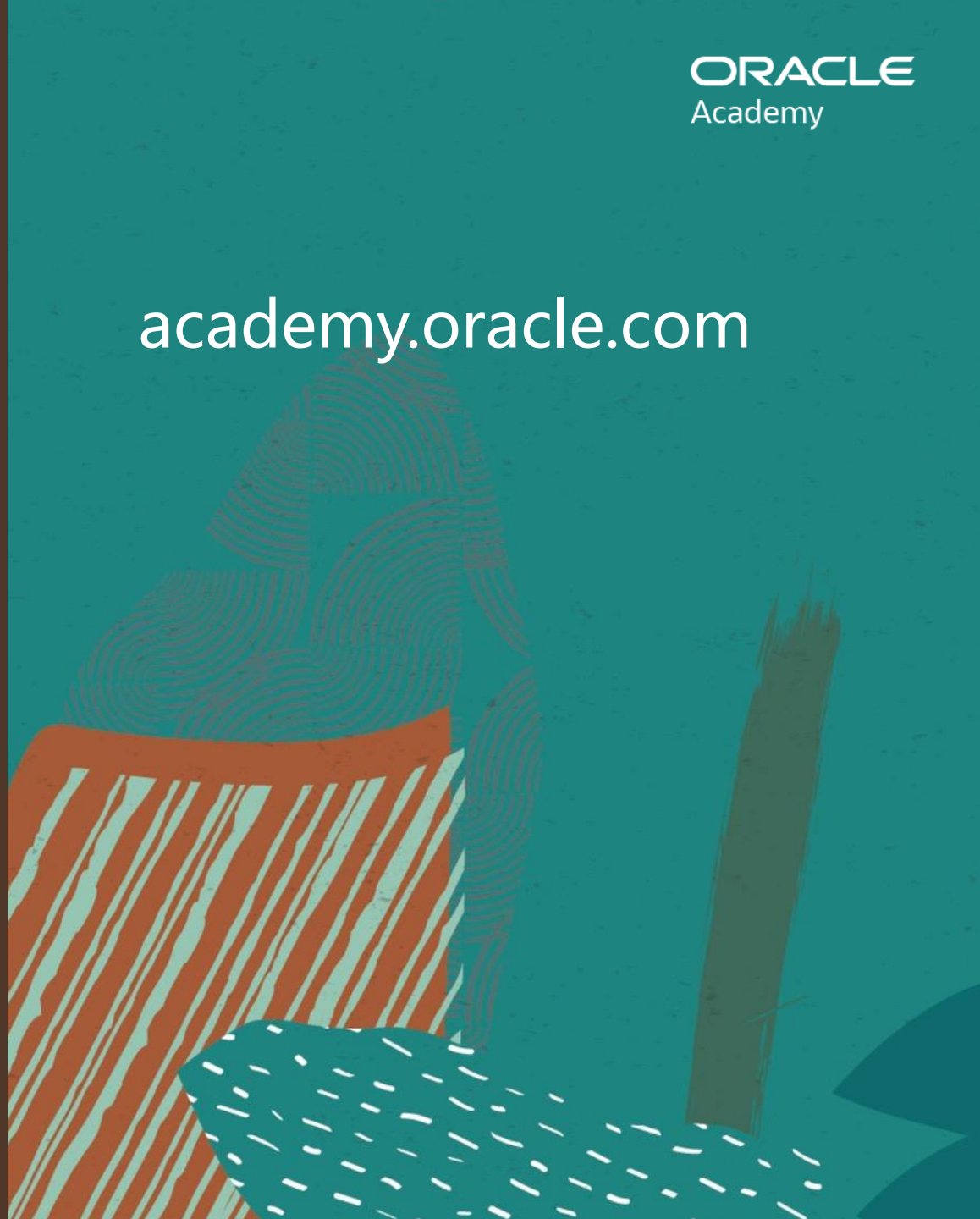
+

Zadaci, kvizevi, projekti

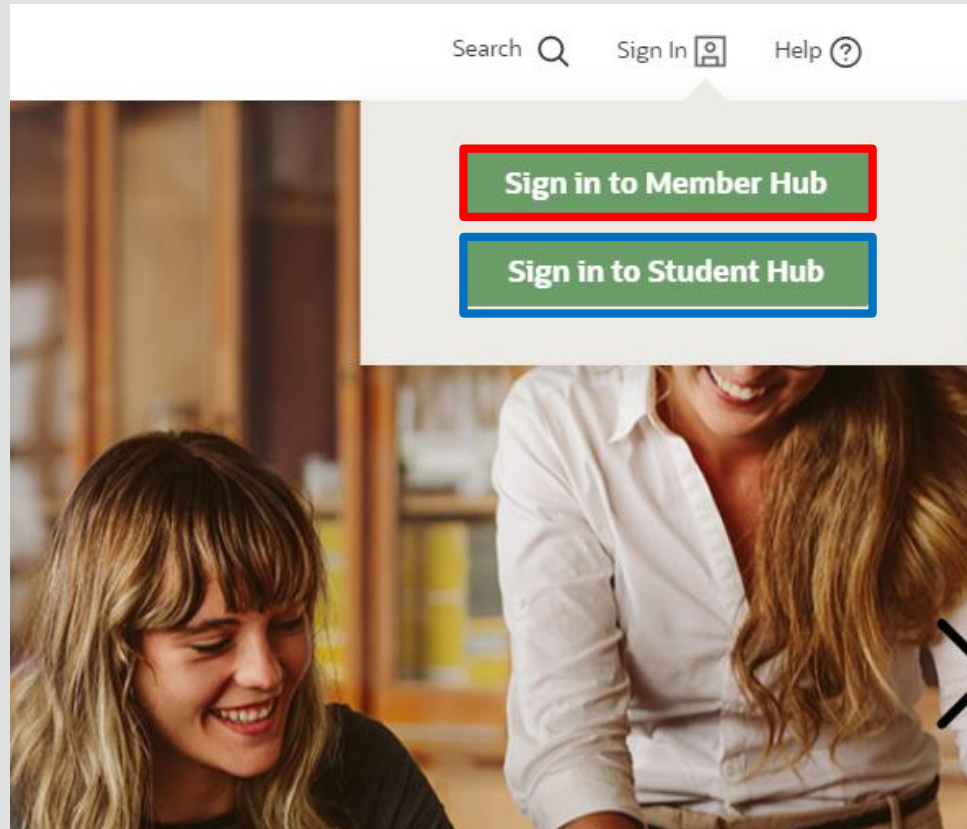
+

Različite radionice

academy.oracle.com

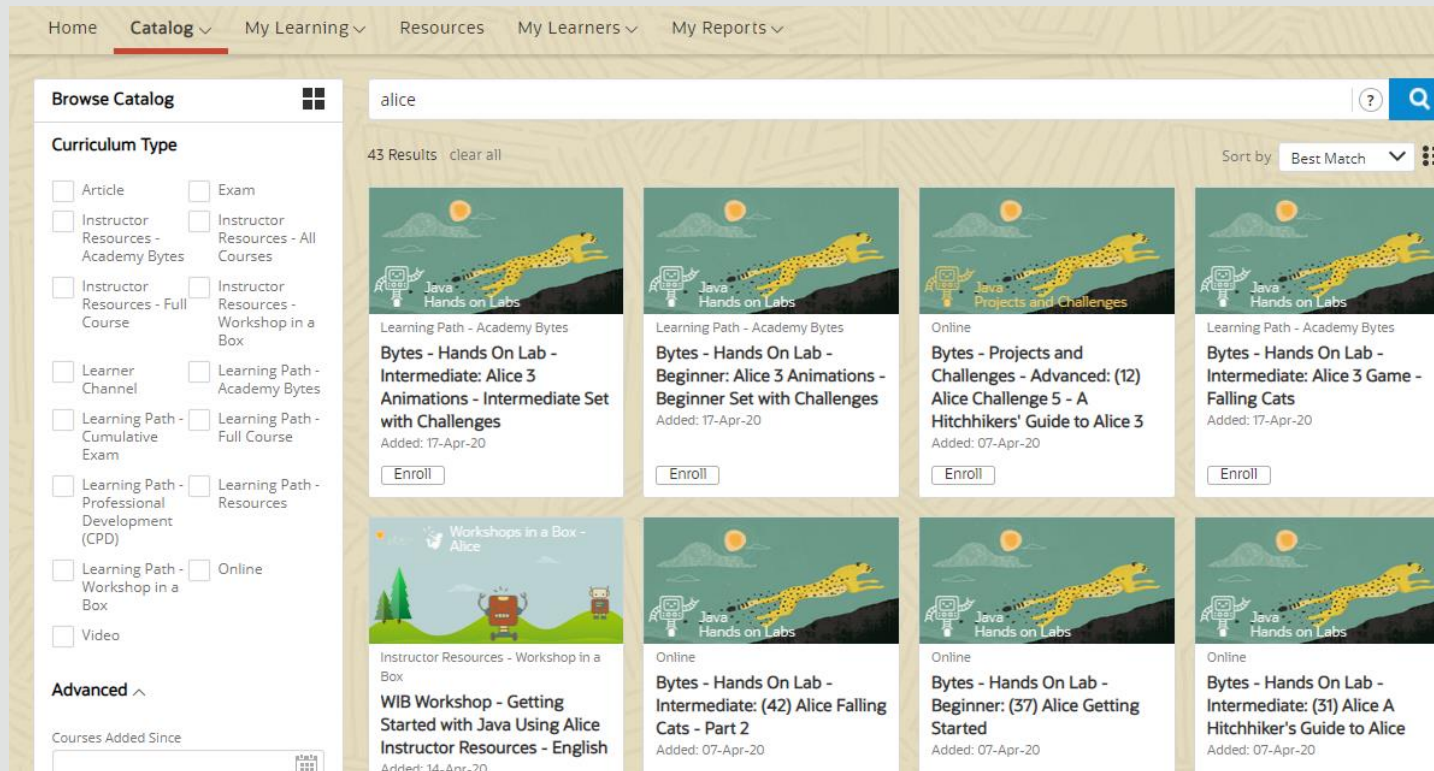


OA Member Hub



- Potpuno novi nastavni resurs aktiviran u sklopu programa početkom Covid 19 krize (uz Oracle Academy Cloud)
- Dva načina prijave :
 - **nastavnici**
 - **učenici**
- Nastavnici izrađuju korisničke račune za učenike

Kako pronaći Alice 3 nastavne materijale ?



- Odabrati izbornik Catalog / Browse
- U retku za pretraživanje upisati Alice

Kako stvoriti predmet (Channel)?

Home
Catalog
My Learning
Resources
My Learners
My Reports

Learning Navigator

Oracle Academy(66327327)

PRODUCTION(66327447)

CURRENT(66327449)

Institutions(68350377)

EMEA(68350379)

Croatia(68350478)

Polytechnic of Sibenik(78419725)

Channels(78419726)

Java Fundamentals - AZOO - 07-01-2020(141374531)

Java Fundamentals - AZOO - 07-01-2020(14137453)

Oracle Academy Alice 07 01 2021(141386224)

Learning Paths(66327453)

Online Courses(66327456)

Channels(78419726)

Properties

Self Paced 1

Create Channel

Channels 1

View

Edit

Targeting

Email Templates

Audit

Folders: 1

Content Objects: 2

Default Content Type:

Online

Created By:

API Admin Account

Modified By:

API Admin Account

Targeting

Condition

Target

Inheritance

Self Enrollment

Polytechnic of Sibenik

PRO

Permissions

Privilege

Target

All

- Odabrati izbornik My Learners / Learning navigator
- Odabrati Create Channel



Izrada korisničkih računa za učenike

The screenshot shows the 'My Learners' interface with the 'Enrollments' tab selected. The interface includes a navigation bar with 'My Learners' and 'My Reports' dropdowns. Below the navigation bar are tabs for 'View', 'Edit', 'Targeting', 'Widgets', 'Enrollments', 'Recommendations', and 'Reference Materials'. The 'Enrollments' tab is active, displaying the following information:

- Identifier: 141374532
- Enrolled / Waitlist / Pending: 5 / - / 0
- Learning Track Only: No
- Language: English

Below this information are two buttons: '+ Add Enrollments' and '+ Create Users And Enroll'. The main section contains search filters:

- User: [Text Input]
- Enrollment Type: [Dropdown Menu with 'Enrolled' selected]
- ☐ Include Expired Users
- Search [Button] Clear [Button]
- Enrollment Status: [Dropdown Menu with 'All' selected]
- Organization: [Text Input]
- User Group: [Text Input]

- Odabrati izbornik My Learners / Learning navigator
- Odabrati predmet (Channel)
- Odabrati Enrollments
- Odabrati Create Users And Enroll

Dodavanje vlastitih nastavnih materijala


Import Content To Channel

1

Choose Upload Type


2

Details



Upload File

- To import a course package, it needs to be zipped up (.zip) and needs to be either an MP4, SCORM, AICC or xAPI course package.
- A SCORM package needs to have a imsmanifest.xml file in the root folder which describes the course and its content.
- An xAPI package needs to have a tincan.xml file in the root folder.
- An AICC package needs to be zipped up with the AICC descriptor files (AU, CRS, CST, & DES).
- A cmi5 package must include a 'cmi5.xml' file which includes a list of assignable units (AU).
- A video file should be MP4 and unzipped. MP4s will use a progressive download player.

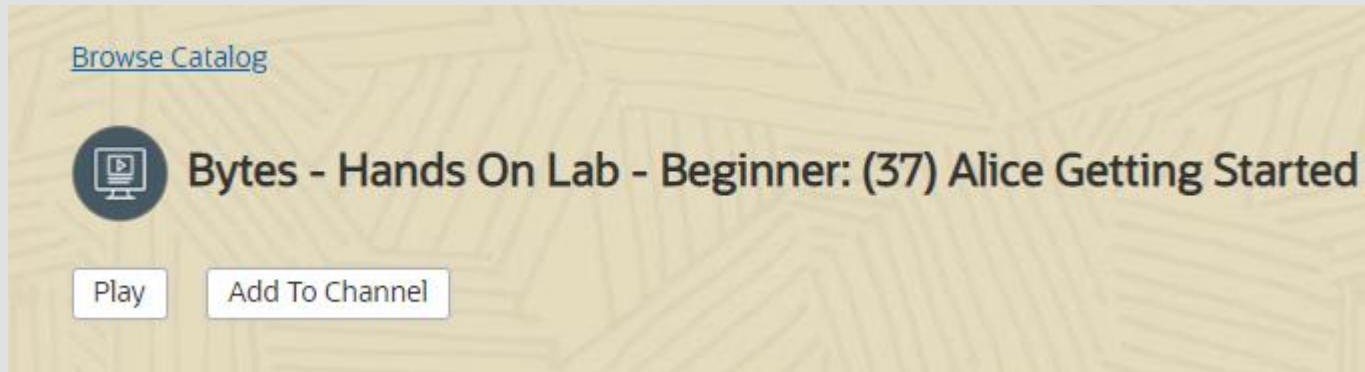


Add External URL

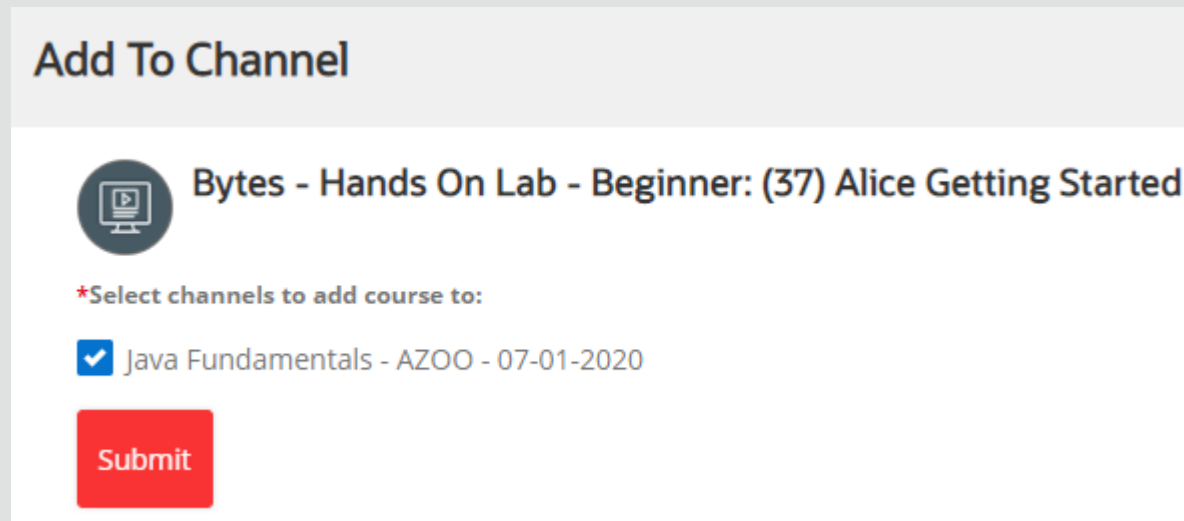
An External URL is a link to any web page or third party content.

- Odabrati izbornik My Learners / Learning navigator
- Odabrati predmet (Channel)
- Odabrati Import Content To Channel

Dodavanje OA nastavnih materijala (Learning Path)



- Odabrati izbornik Catalog / Browse
- Pronaći nastavni materijal
- Add to Channel
- Odabrati predmet (Channel) i potvrditi sa Submit



Praćenje učenika

[Reports](#)

Learner Channel Status Report (65280)

Lists all channel learners within your institution and their status in the objects in that channel.

1. Locate your Channel from Channel Name list-of-values.
2. Select Run button.

Report Format:

* Channel Name:

* Mandatory fields


- Odabrati izbornik My Reports / naredba My Reports
- Odabrati Learner Channel Status Report
- Zadati brojčanu oznaku predmeta
- Otvoriti izvješće sa naredbom Run


Što se očekuje u sljedećoj
verziji ?

Alice 3 Unity + VR


Downloads & Useful Links


Alice 3 Unity Player

**Windows**
Unity Player
Size: 78.9 MB


**Mac**
Unity Player
Size: 84.9 MB


Alice 3 IDE Beta Builds


**Windows (64 Bit)**
Beta Installer
Size: 1.45 GB


**Mac**
Beta Installer
Size: 1.47 GB


Alice Player Demo Worlds


**CSTA and ISTE Demos**
ZIP Folder
Size: 65.6 MB

 **Using The Alice Player**

 **Exporting For The Alice Player**

 **Using The Alice Player With Oculus VR**

 **Using The Alice Player With Vive VR**

 **Overview of VR Input Mapping**

Pitanja ?

Pitanja

- Q: Kako se učlaniti u OA program ?
- A: Putem poveznice academy.oracle.com, odaberite Join now
- Q: Alice 3 priručnik na hrvatskom jeziku ?
- A: U izradi, očekivani završetak listopad 2021. godine
(za sada dostupno na poveznici: https://vushr-my.sharepoint.com/:u:/g/personal/frane_vus_hr/EQmOfoTRSOINnUAcO1FkeTQBca3NYUIOrUYgid_-XraT6g?e=GeRy5c)
- Q: Gdje mogu pronaći danas prikazane Alice 3 primjere ?
- A: Na poveznici: https://vushr-my.sharepoint.com/:u:/g/personal/frane_vus_hr/EUaEiKMKnjdlihsmEjtCqcMBQaLiAY7Sh9_I9WRRFKWSYw?e=wf3P3r
- Q: Dodatna pitanja ?
- A: Slobodno pošaljite upit na darko.jurekovic@oracle.com ili frane.urem@oracle.com



Hvala na pažnji !

academy.oracle.com

