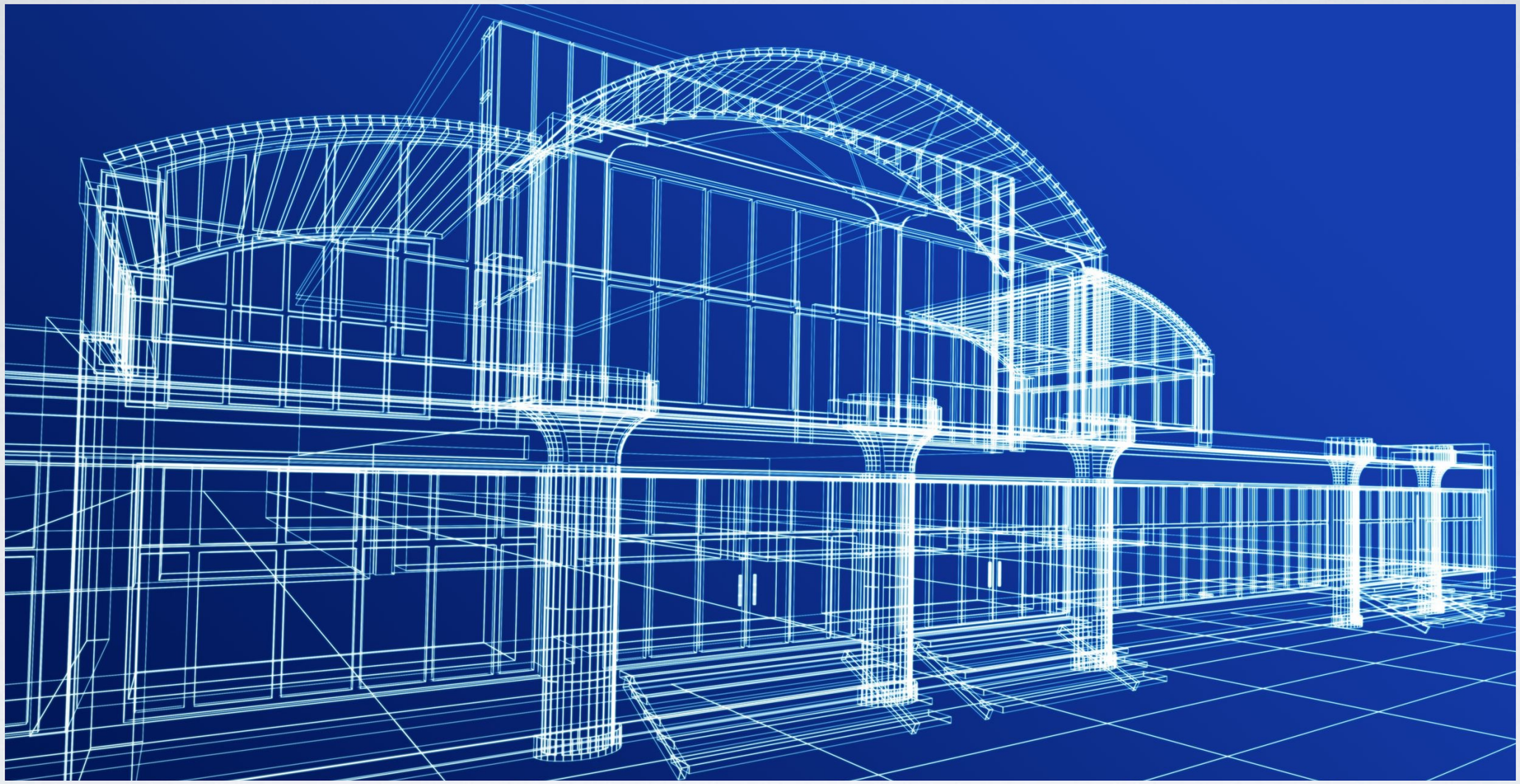
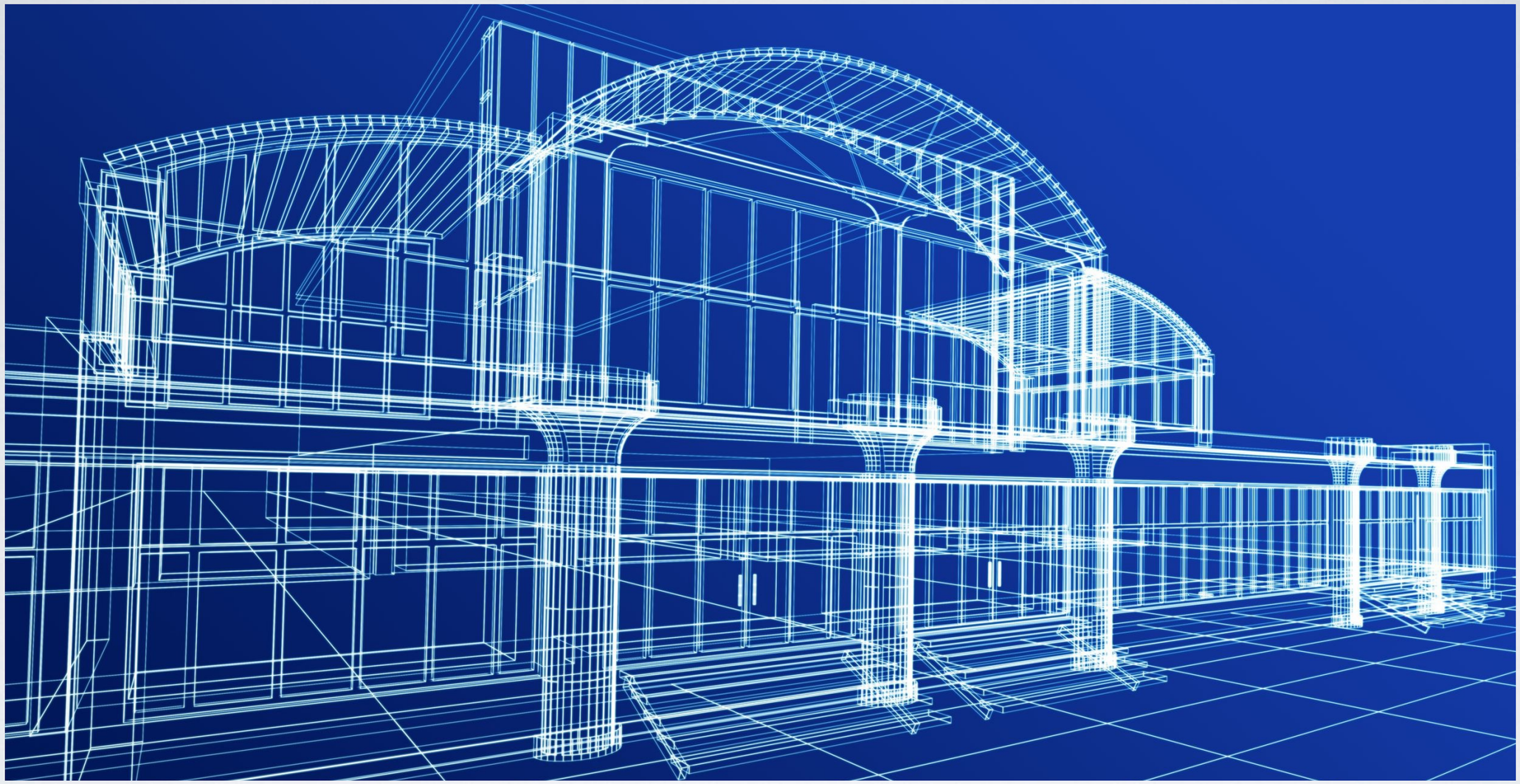
THINK LIKE AN ARCHITECT:
TYPES

Lesson 8



THINK ABOUT DIFFERENT KINDS OF BUILDINGS



THINK ABOUT DIFFERENT KINDS OF BUILDINGS

LET’S GET STARTED

- Name different types of buildings.
- Let’s pick one type to think about.
- In Notes, write down what makes this type of building unique? What are some specific properties that helps you identify it?

LET’S GET STARTED

- Name different types of buildings.
- Let’s pick one type to think about.
- In Notes, write down what makes this type of building unique? What are some specific properties that helps you identify it?

LET’S DISCUSS

1. What properties did you think of?
2. Do we all agree that these are the properties? Are there any missing? Any that we shouldn’t include?
3. Are each of the properties clear? Would we be able to build this building based on our list?

LET’S DISCUSS

1. What properties did you think of?
2. Do we all agree that these are the properties? Are there any missing? Any that we shouldn’t include?
3. Are each of the properties clear? Would we be able to build this building based on our list?

IMPORTANT VOCABULARY

Type: A named grouping of properties (the features) and methods (the behaviors) of a kind of data.

Initialization: The act of creating a new instance of a type, which includes setting initial values for any properties of the type.

IMPORTANT VOCABULARY

Type: A named grouping of properties (the features) and methods (the behaviors) of a kind of data.

Initialization: The act of creating a new instance of a type, which includes setting initial values for any properties of the type.

IMPORTANT VOCABULARY

Here are a few different data types:

String: A type that stores a series for characters, such as “hello world.”

Int: A type that stores an integer - a number that has no decimal, such as 10 or -42.

Bool: A type that stores a value of either “true” or “false.”

IMPORTANT VOCABULARY

Here are a few different data types:

String: A type that stores a series for characters, such as “hello world.”

Int: A type that stores an integer - a number that has no decimal, such as 10 or -42.

Bool: A type that stores a value of either “true” or “false.”

BE AN ARCHITECT

1. Choose a type of building. This can be real or imagined.

2. In Notes, write down the variables your building will have.

    - ex. numberOfWindows

BE AN ARCHITECT

1. Choose a type of building. This can be real or imagined.

2. In Notes, write down the variables your building will have.

    - ex. numberOfWindows

BE AN ARCHITECT

3. Now add the values next to the variables. This step describes a specific instance of the building type. So you are now initializing an instance of your house type. You should indicate where you used string, int, and bool data types.
    - ex. numberOfWindows = 8
4. In Notes, or another drawing app, draw your building using your defined variables.

BE AN ARCHITECT

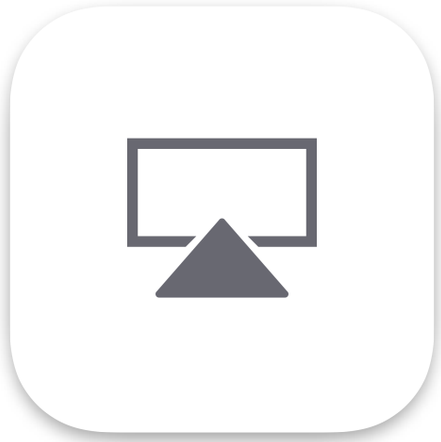
3. Now add the values next to the variables. This step describes a specific instance of the building type. So you are now initializing an instance of your house type. You should indicate where you used string, int, and bool data types.
    - ex. numberOfWindows = 8
4. In Notes, or another drawing app, draw your building using your defined variables.

LET’S TEST

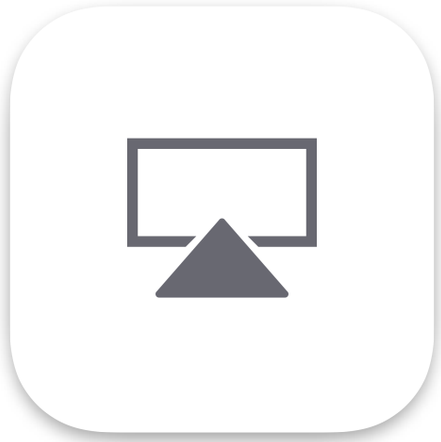
1. Find a partner.
2. Share only your text with your type and instance values with your partner.
3. Draw your partner’s building.
4. Share your drawings. How similar do they look?

LET’S TEST

1. Find a partner.
2. Share only your text with your type and instance values with your partner.
3. Draw your partner’s building.
4. Share your drawings. How similar do they look?



Share your types, instance values,
and drawings.



Share your types, instance values,
and drawings.

LET’S DISCUSS

1. How similar were you and your partner’s drawings?

2. How can you make them more alike?

LET’S DISCUSS

1. How similar were you and your partner’s drawings?

2. How can you make them more alike?

TIME FOR SWIFT PLAYGROUNDS

Chapters: Types and Initialization

Types	
Introduction	✓
Deactivating a Portal	✓
Portal On and Off	✓
Setting the Right Portal	✓
Corners of the World	✓
Random Gems Everywhere	✓
Initialization	
Introduction	✓
Initializing Your Expert	✓
Train Your Expert	✓
Using Instances of Different Types	✓
It Takes Two	✓

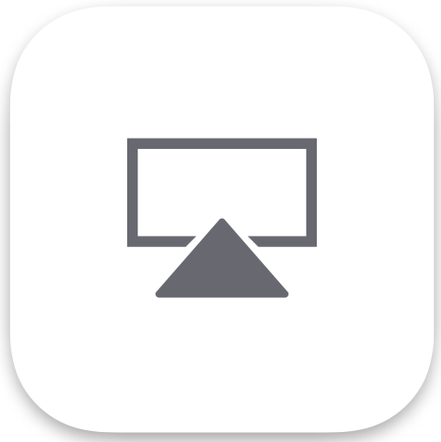
REMINDER: Take videos and or photos of your playgrounds. You will need them for your portfolio.

TIME FOR SWIFT PLAYGROUNDS

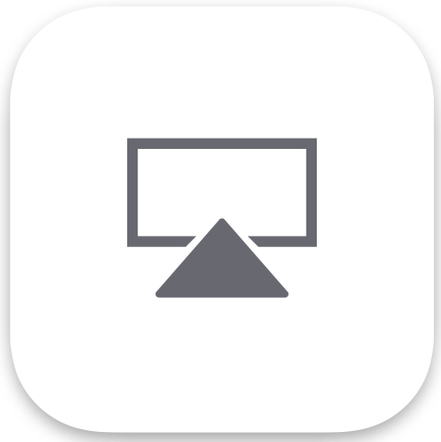
Chapters: Types and Initialization

Types	
Introduction	✓
Deactivating a Portal	✓
Portal On and Off	✓
Setting the Right Portal	✓
Corners of the World	✓
Random Gems Everywhere	✓
Initialization	
Introduction	✓
Initializing Your Expert	✓
Train Your Expert	✓
Using Instances of Different Types	✓
It Takes Two	✓

REMINDER: Take videos and or photos of your playgrounds. You will need them for your portfolio.



Share what you did in Swift Playgrounds with AirPlay



Share what you did in Swift Playgrounds with AirPlay

LET'S REFLECT

1. What were the types in the app?
2. What did you initialize?
3. How was the code you wrote in the app similar or different from the code you wrote for your building?
4. Do types in everyday life differ from types in coding? Why or why not?

Think ahead: What other ways can you provide more detail in code?

LET'S REFLECT

1. What were the types in the app?
2. What did you initialize?
3. How was the code you wrote in the app similar or different from the code you wrote for your building?
4. Do types in everyday life differ from types in coding? Why or why not?

Think ahead: What other ways can you provide more detail in code?

JOURNAL

1. Upload your type, instance values, and drawings.
2. Upload videos and photos from Swift Playgrounds.
3. Record answers to these questions:

    - What is type and initialization?
      (Use your own words.)
    - What do you know about thinking like an architect who is also a programmer?

JOURNAL

1. Upload your type, instance values, and drawings.
2. Upload videos and photos from Swift Playgrounds.
3. Record answers to these questions:

    - What is type and initialization?
      (Use your own words.)
    - What do you know about thinking like an architect who is also a programmer?