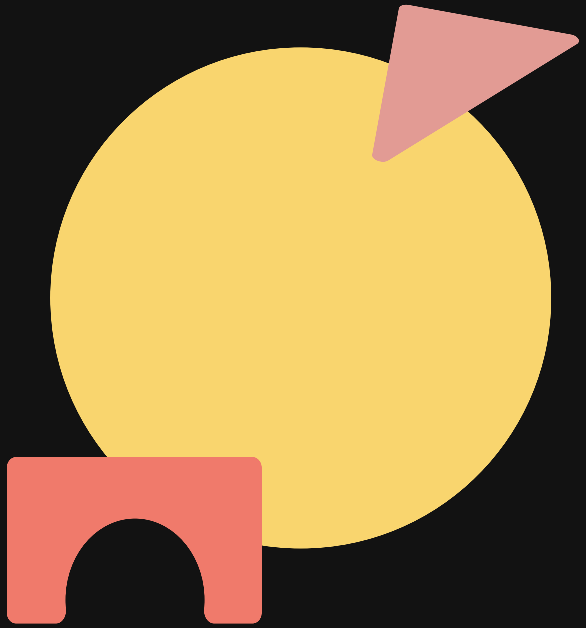
Lightweight Modeling with PlantUML

오상문

Oct. 2023

ToC

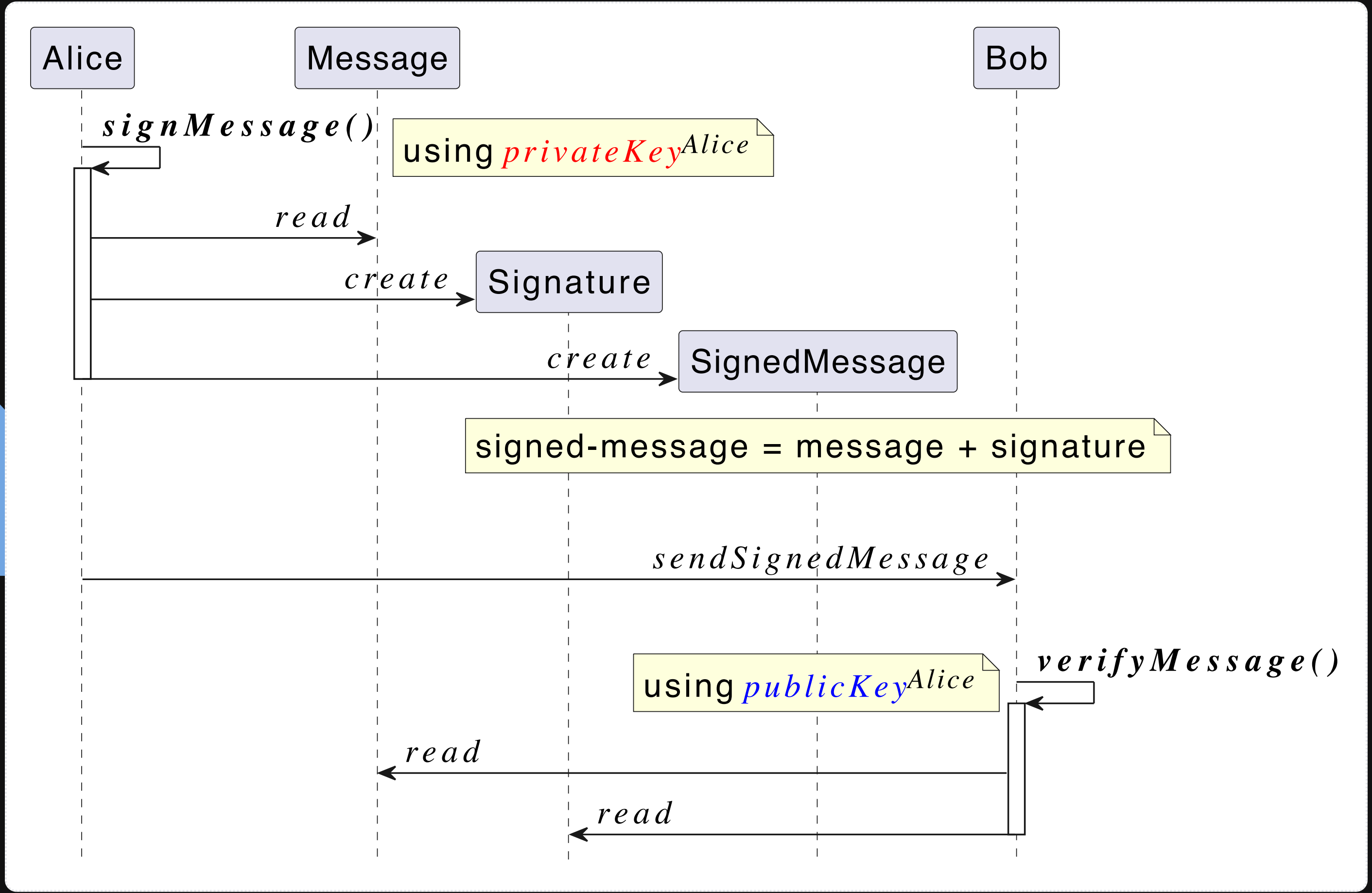


Intro

Get this from

https://github.com/3rdstage/guides

Sample Diagram 1

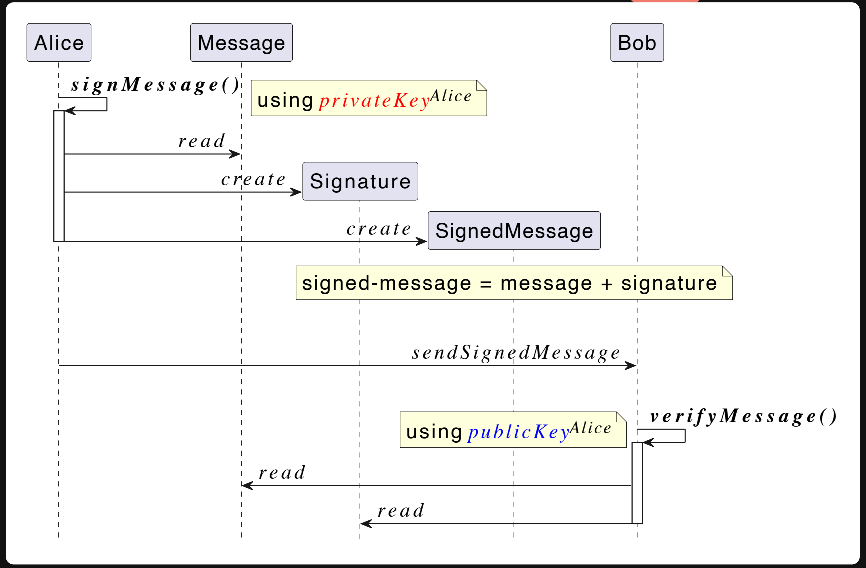


- Actors
- Participants
- Lifelines
- Messages
- Activations
- Combined Fragments
- Notes

- References
  - What is Sequence Diagram?
  - UML Quick Reference
  - UML 2.5.1 Specifiation

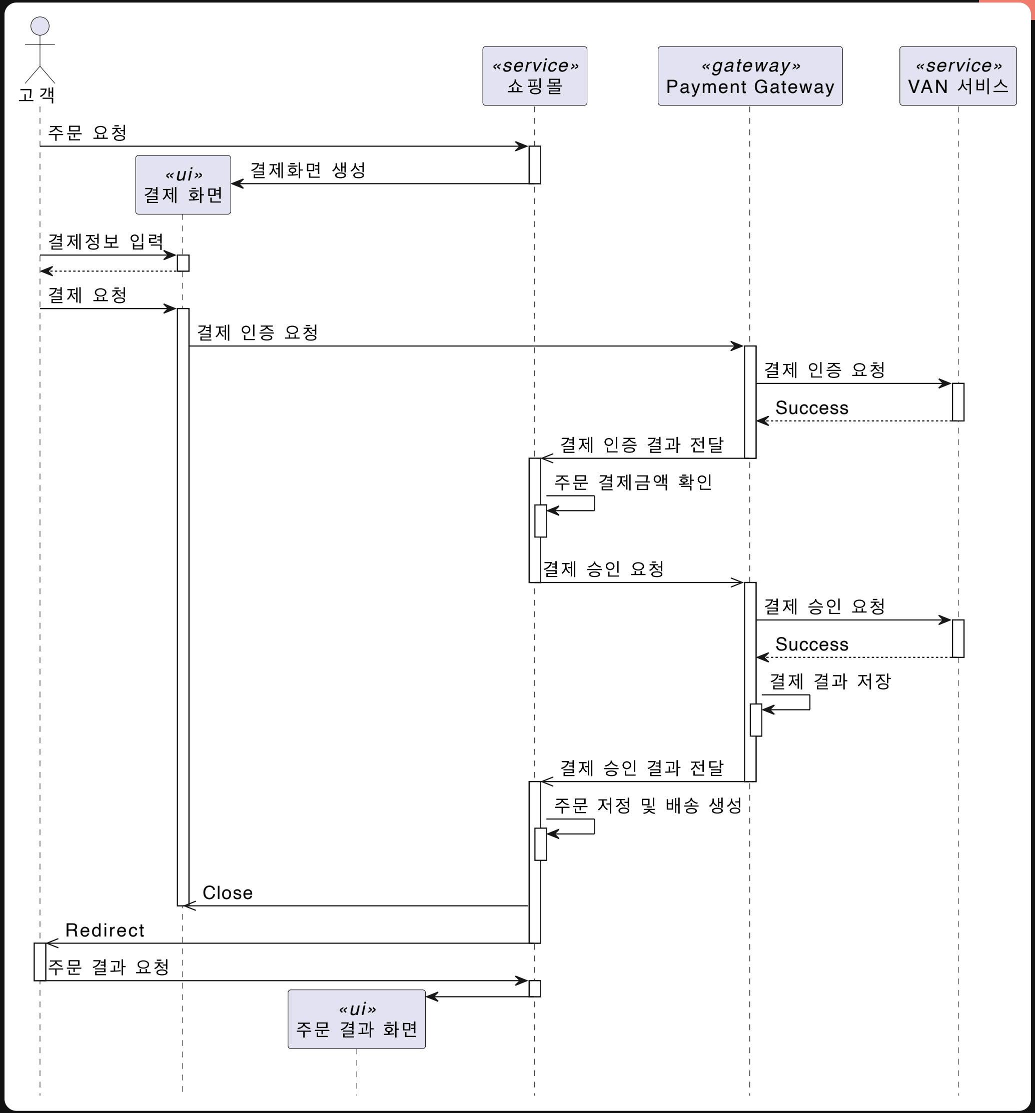
First PlantUML

```
@startuml

skinparam DefaultFontSize 20
skinparam DefaultMonospacedFontName Consolas
skinparam ArrowFontName Consolas
skinparam ArrowFontStyle italic
skinparam SequenceMessageAlign reverseDirection
hide FootBox

Alice -> Alice ++ : **signMessage()**
  note right : using //""<color:red>privateKey</color><sup>Alice</su
  Alice -> Message : read
  Alice -> Signature ** : create
  Alice -> SignedMessage ** : create
    note over SignedMessage : signed-message = message + signature
  Alice --

||30||
Alice -> Bob  : sendSignedMessage
||30||

Bob -> Bob ++ : **verifyMessage()**
  note left : using //""<color:blue>publicKey</color><sup>Alice</sup
  Bob -> Message : read
  Bob -> Signature -- : read
@enduml
```

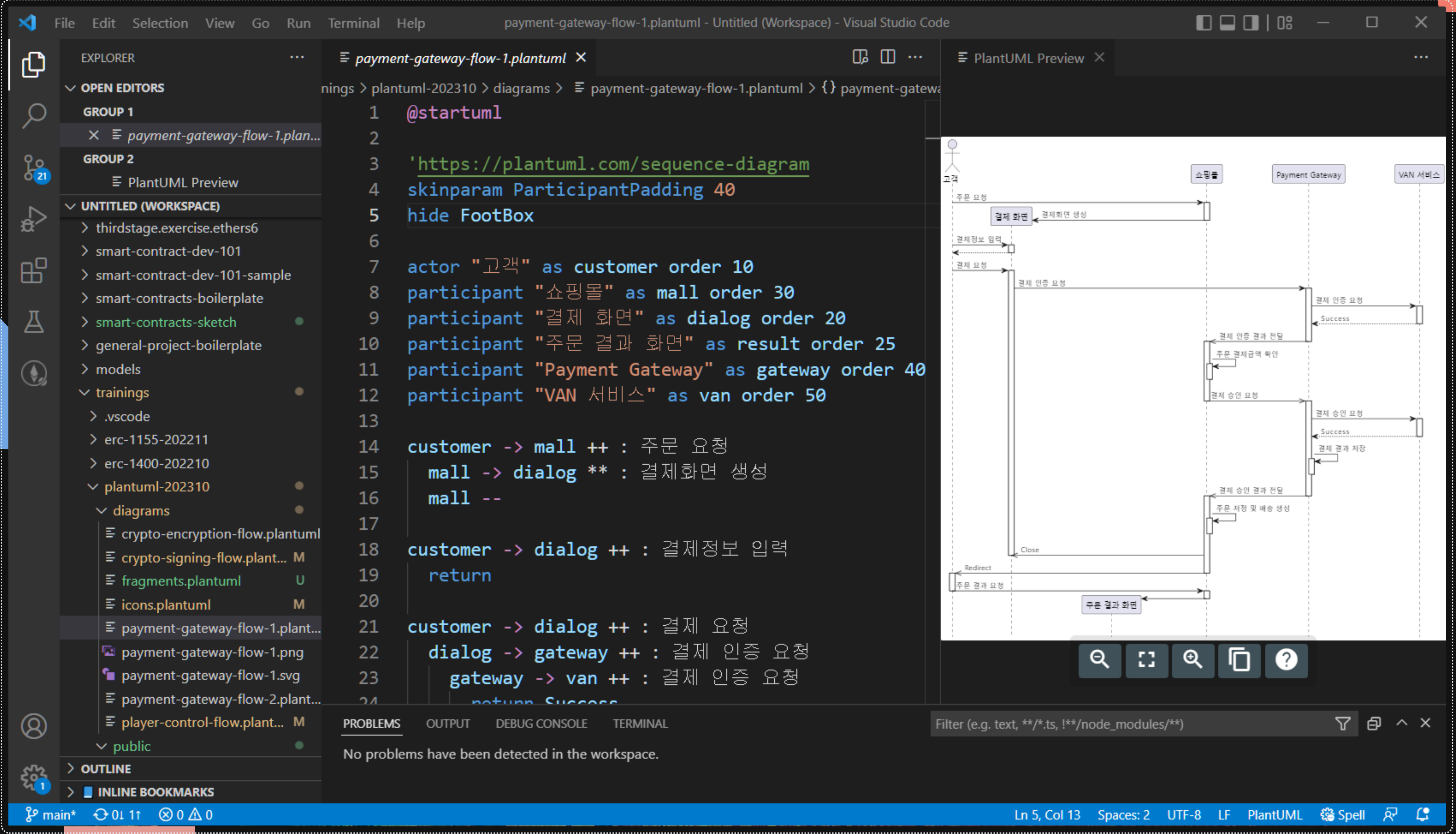


Sample Diagram 2

```
@startuml

' https://plantuml.com/en/sequence-diagram
skinparam DefaultFontSize 16
skinparam ParticipantPadding 20
hide FootBox

actor "고객" as customer order 10
participant "쇼핑몰" as mall <<service>> order 30
participant "결제 화면" as dialog <<ui>> order 20
participant "주문 결과 화면" as result <<ui>> order 25
participant "Payment Gateway" as gateway <<gateway>> order 40
participant "VAN 서비스" as van <<service>> order 50

customer -> mall ++ : 주문 요청
  mall -> dialog ** : 결제화면 생성
  mall --

customer -> dialog ++ : 결제정보 입력
  return

customer -> dialog ++ : 결제 요청
  dialog -> gateway ++ : 결제 인증 요청
    gateway -> van ++ : 결제 인증 요청
      return Success
    gateway ->> mall --++ : 결제 인증 결과 전달
      mall -> mall ++-- : 주문 결제금액 확인
      mall ->> gateway --++ : 결제 승인 요청
        gateway -> van ++ : 결제 승인 요청
          return Success
        gateway -> gateway ++-- : 결제 결과 저장
        gateway ->> mall --++ : 결제 승인 결과 전달
          mall -> mall ++-- : 주문 저장 및 배송 생성
          mall  ->> dialog : Close
            dialog --
          mall ->> customer --++ : Redirect
            customer -> mall --++ : 주문 결과 요청
              mall -> result **
              mall --
||20||
@enduml
```



VS Code PlantUML Extension

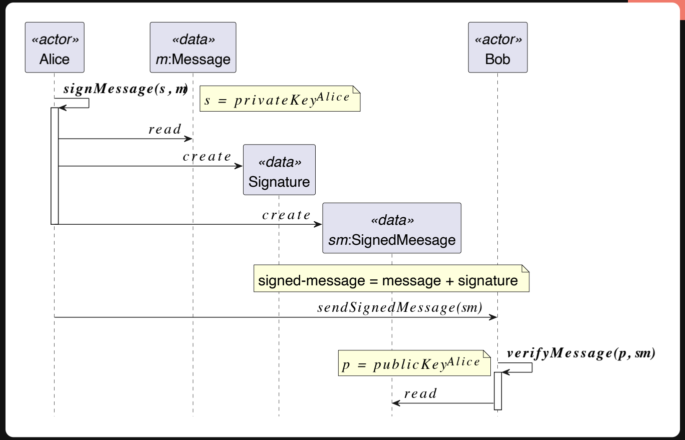


```
@startuml

'https://plantuml.com/sequence-diagram
skinparam ParticipantPadding 40
hide FootBox

actor "고객" as customer order 10
participant "쇼핑몰" as mall order 30
participant "결제 화면" as dialog order 20
participant "주문 결과 화면" as result order 25
participant "Payment Gateway" as gateway order 40
participant "VAN 서비스" as van order 50

customer -> mall ++ : 주문 요청
  mall -> dialog ** : 결제화면 생성
  mall --

customer -> dialog ++ : 결제정보 입력
  return

customer -> dialog ++ : 결제 요청
  dialog -> gateway ++ : 결제 인증 요청
    gateway -> van ++ : 결제 인증 요청
      return Success
```

No problems have been detected in the workspace.

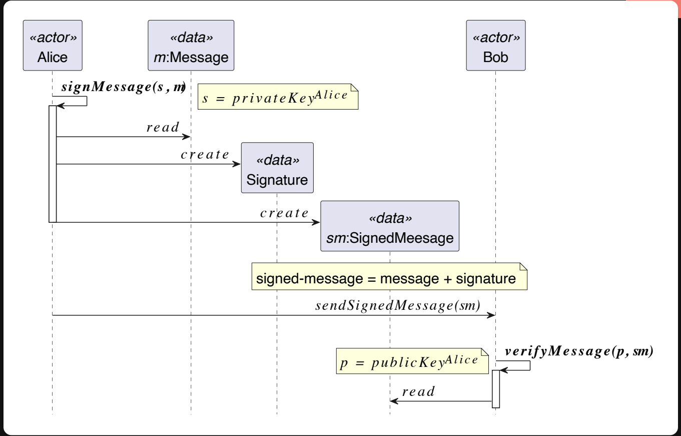
Sequence Diagram - Step by Step 1/3

```
@startuml

' https://plantuml.com/en/sequence-diagram
skinparam DefaultFontSize 20
skinparam DefaultMonospacedFontName Consolas
skinparam ArrowFontName Consolas
skinparam ArrowFontStyle italic
skinparam SequenceMessageAlign reverseDirection
hide FootBox

participant Alice as alice <<actor>> order 10
participant Bob as bob <<actor>> order 40
participant "//m//:Message" as msg <<data>> order 15
participant Signature as sig <<data>> order 20
participant "//sm//:SignedMeesage" as signed <<data>> order 30

alice -> alice ++ : **signMessage(//s//,//m//)**
  note right : //""s = privateKey<sup>Alice</sup>""//
  alice -> msg : read
  alice -> sig ** : create
  alice -> signed ** : create
    note over signed
      signed-message = message + signature
    end note
  alice --

alice -> bob  : sendSignedMessage(//sm//)
||30||

bob -> bob ++ : **verifyMessage(//p//,//sm//)**
  note left : //""p = publicKey<sup>Alice</sup>""//
  bob -> signed : read
@enduml
```



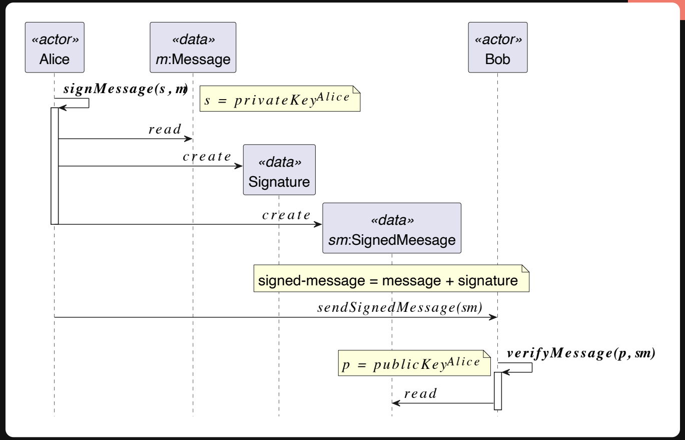
Sequence Diagram - Step by Step 2/3

```
@startuml

' https://plantuml.com/en/sequence-diagram
skinparam DefaultFontSize 20
skinparam DefaultMonospacedFontName Consolas
skinparam ArrowFontName Consolas
skinparam ArrowFontStyle italic
skinparam SequenceMessageAlign reverseDirection
hide FootBox

participant Alice as alice <<actor>> order 10
participant Bob as bob <<actor>> order 40
participant "//m//:Message" as msg <<data>> order 15
participant Signature as sig <<data>> order 20
participant "//sm//:SignedMeesage" as signed <<data>> order 30

alice -> alice ++ : **signMessage(//s//,//m//)**
  note right : //""s = privateKey<sup>Alice</sup>""//
  alice -> msg : read
  alice -> sig ** : create
  alice -> signed ** : create
    note over signed
      signed-message = message + signature
    end note
  alice --

alice -> bob  : sendSignedMessage(//sm//)
||30||

bob -> bob ++ : **verifyMessage(//p//,//sm//)**
  note left : //""p = publicKey<sup>Alice</sup>""//
  bob -> signed : read
@enduml
```



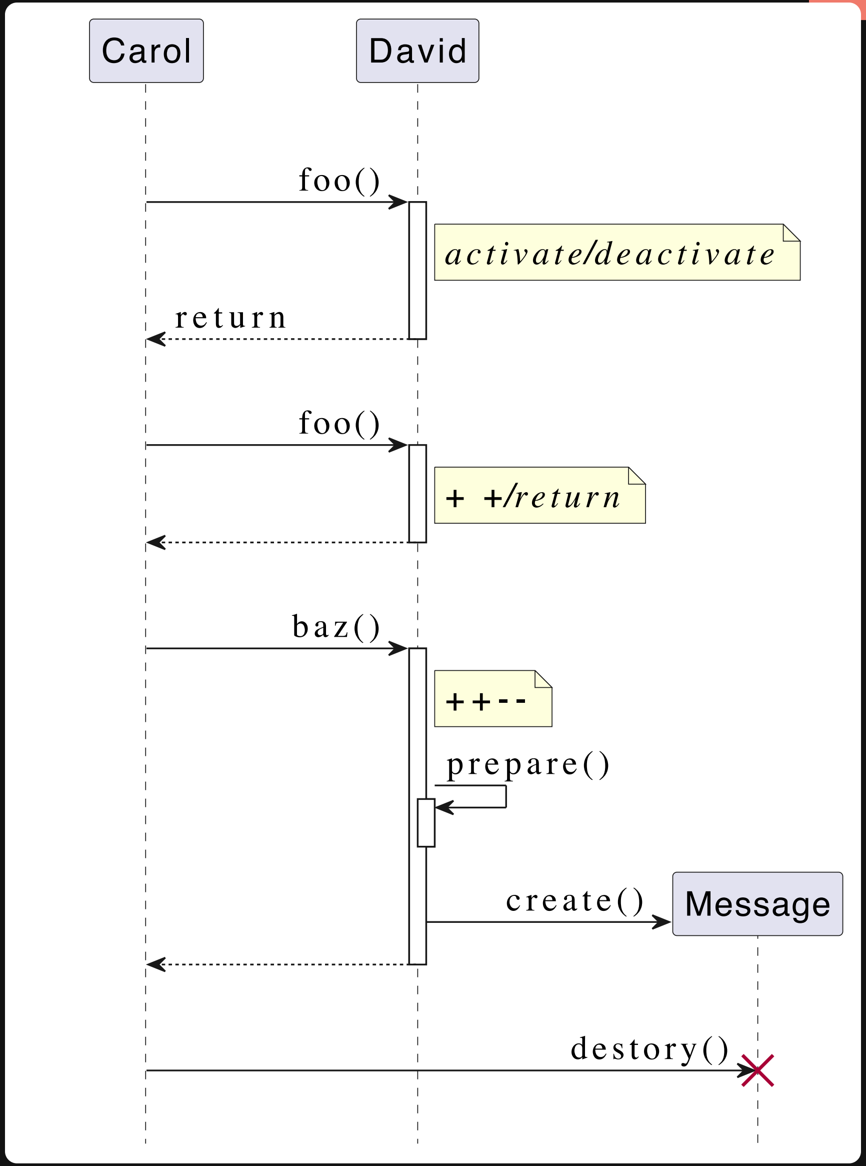
Sequence Diagram - Step by Step 3/3

```
@startuml

' https://plantuml.com/en/sequence-diagram
skinparam DefaultFontSize 20
skinparam DefaultMonospacedFontName Consolas
skinparam ArrowFontName Consolas
skinparam ArrowFontStyle italic
skinparam SequenceMessageAlign reverseDirection
hide FootBox

participant Alice as alice <<actor>> order 10
participant Bob as bob <<actor>> order 40
participant "//m//:Message" as msg <<data>> order 15
participant Signature as sig <<data>> order 20
participant "//sm//:SignedMeesage" as signed <<data>> order 30

alice -> alice ++ : **signMessage(//s//,//m//)**
  note right : //""s = privateKey<sup>Alice</sup>""//
  alice -> msg : read
  alice -> sig ** : create
  alice -> signed ** : create
    note over signed
      signed-message = message + signature
    end note
  alice --

alice -> bob  : sendSignedMessage(//sm//)
||30||

bob -> bob ++ : **verifyMessage(//p//,//sm//)**
  note left : //""p = publicKey<sup>Alice</sup>""//
  bob -> signed : read
@enduml
```



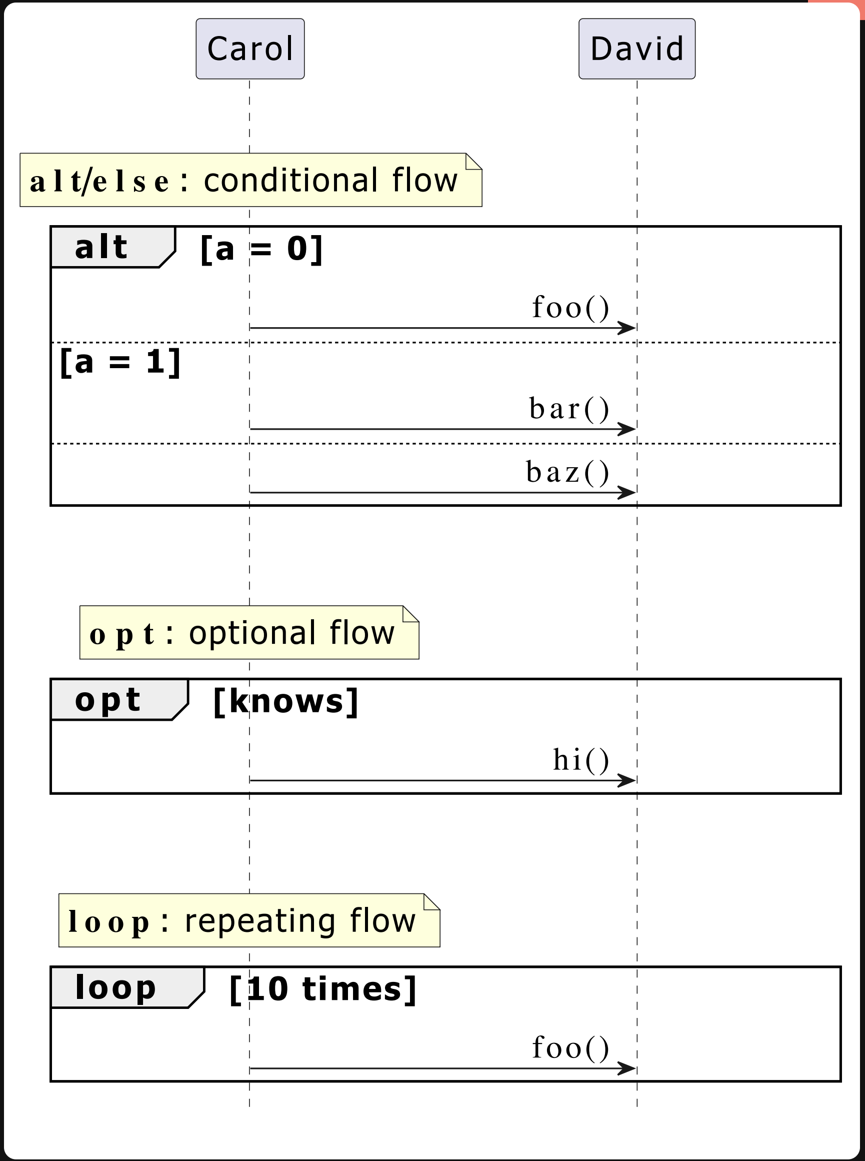
Sequence Diagram - Activations

```
@startuml

skinparam DefaultFontSize 20
skinparam DefaultMonospacedFontName Consolas
skinparam ArrowFontName Consolas
skinparam ParticipantPadding 40
skinparam SequenceMessageAlign reverseDirection
hide FootBox

participant Carol as carol
participant David as david
participant Message as msg

||30||
carol -> david : foo()
  note right of david : //""activate""/////""deactivate""//
  activate david
  carol <-- david : return
  deactivate david
|||
carol -> david ++ : foo()
  note right of david : ++///""return""//
  return
|||
carol -> david ++ : baz()
  note right of david : ++--
  david -> david ++-- : prepare()
  david -> msg ** : create()
  return
|||
carol -> msg !! : destory()
|||
@enduml
```



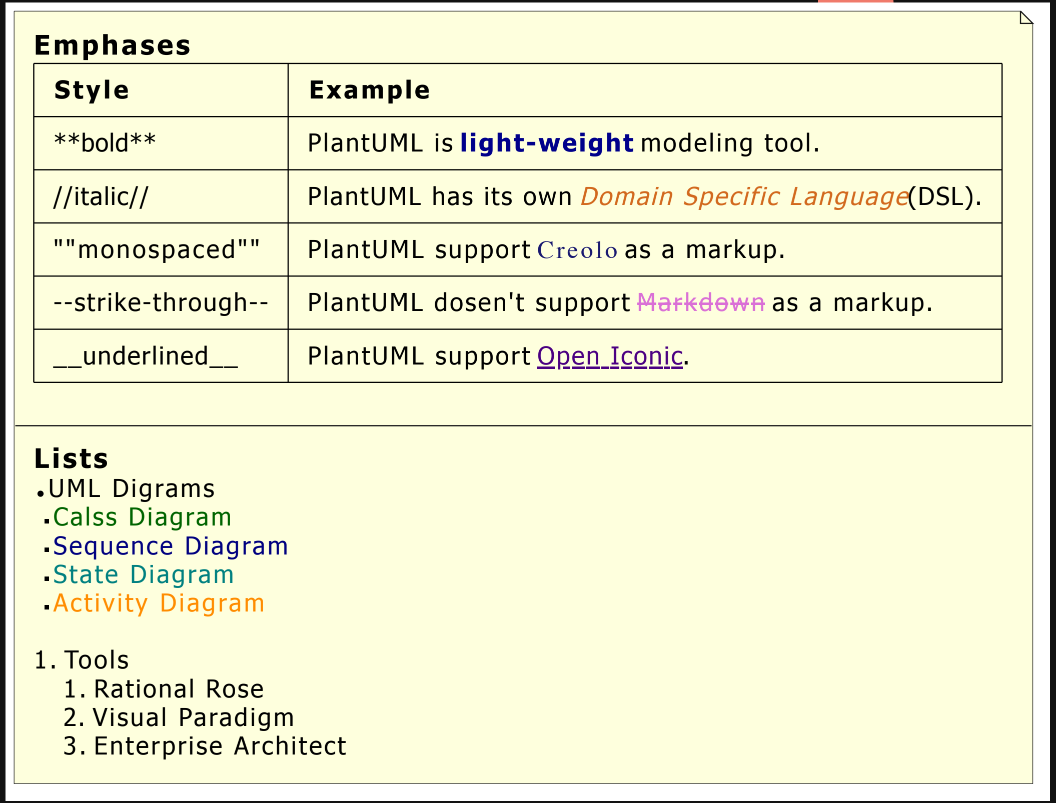
Sequence Diagram - Fragments

```
@startuml

skinparam DefaultMonospacedFontName Consolas
skinparam ArrowFontName Consolas
skinparam ParticipantPadding 80
skinparam SequenceMessageAlign reverseDirection
hide FootBox

participant Carol as carol
participant David as david

||30||
note over carol : ""**alt**""/""**else**"" : conditional flow
alt a = 0
  carol -> david : foo()
else a = 1
  carol -> david : bar()
else
  carol -> david : baz()
end

||50||
note over carol : ""**opt**"" : optional flow
opt knows
  carol -> david : hi()
end

||50||
note over carol : ""**loop**"" : repeating flow
loop 10 times
  carol -> david : foo()
end
@enduml
```



Markup and Colors

```
@startuml
skinparam DefaultFontSize 20
skinparam DefaultFontName Tahoma
skinparam DefaultMonospacedFontName Consolas
skinparam Padding 10

note as creole
== Emphases
|= Style |= Example |
| ~**bold**  | PlantUML is <color:darkblue>**light-weight**</c
| ~//italic// | PlantUML has its own <color:chocolate>//Domain
| ~""monospaced"" | PlantUML support <color:midnightblue>""Cre
| ~--strike-through-- | PlantUML dosen't support <color:orchid
| ~__underlined__ | PlantUML support <color:indigo>__Open Icon

----
== Lists
* UML Digrams
** <color:#006400>Calss Diagram</color>
** <color:navy>Sequence Diagram</color>
** <color:teal>State Diagram</color>
** <color:darkorange>Activity Diagram</color>

# Tools
## Rational Rose
## Visual Paradigm
## Enterprise Architect
end note
@enduml
```



Emphases

Style	**Example**
bold	PlantUML is **light-weight** modeling tool.
//italic//	PlantUML has its own *Domain Specific Language*(DSL).
""monospaced""	PlantUML support `Creolo` as a markup.
--strike-through--	PlantUML dosen't support ~~Markdown~~ as a markup.
__underlined__	PlantUML support Open Iconic.

Lists

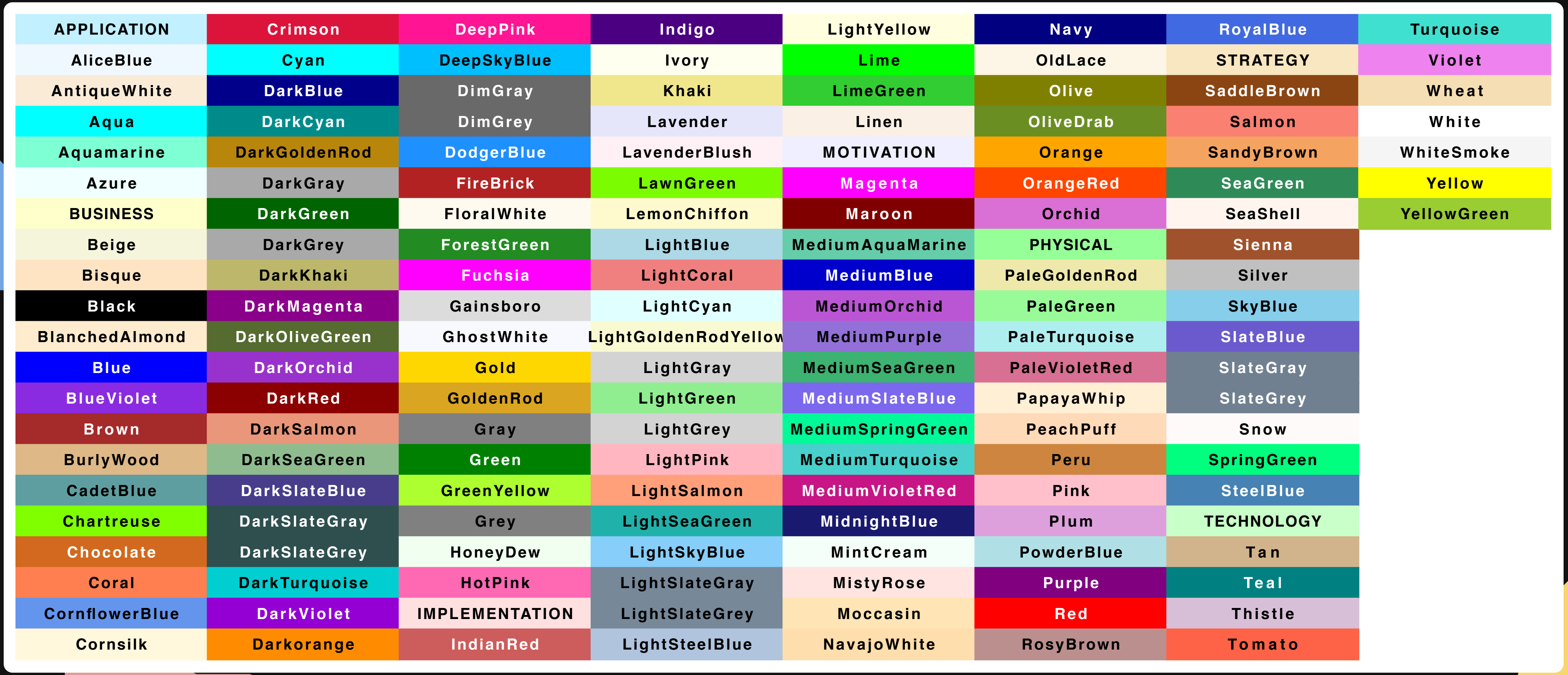
- UML Digrams
  - Calss Diagram
  - Sequence Diagram
  - State Diagram
  - Activity Diagram

1. Tools
   1. Rational Rose
   2. Visual Paradigm
   3. Enterprise Architect

- References
  - PlantUML / Creole
  - PlantUML / Colors

Color Palette

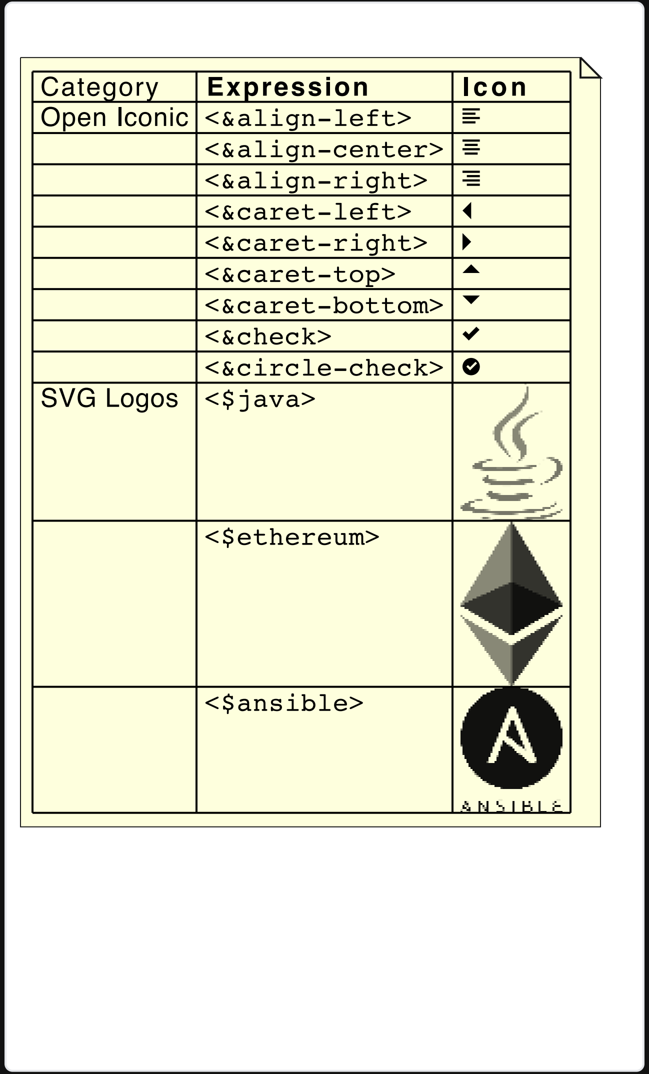
```
@startuml
colors
@enduml
```



APPLICATION	Crimson	DeepPink	Indigo	LightYellow	Navy	RoyalBlue	Turquoise
AliceBlue	Cyan	DeepSkyBlue	Ivory	Lime	OldLace	STRATEGY	Violet
AntiqueWhite	DarkBlue	DimGray	Khaki	LimeGreen	Olive	SaddleBrown	Wheat
Aqua	DarkCyan	DimGrey	Lavender	Linen	OliveDrab	Salmon	White
Aquamarine	DarkGoldenRod	DodgerBlue	LavenderBlush	MOTIVATION	Orange	SandyBrown	WhiteSmoke
Azure	DarkGray	FireBrick	LawnGreen	Magenta	OrangeRed	SeaGreen	Yellow
BUSINESS	DarkGreen	FloralWhite	LemonChiffon	Maroon	Orchid	SeaShell	YellowGreen
Beige	DarkGrey	ForestGreen	LightBlue	MediumAquaMarine	PHYSICAL	Sienna	
Bisque	DarkKhaki	Fuchsia	LightCoral	MediumBlue	PaleGoldenRod	Silver	
Black	DarkMagenta	Gainsboro	LightCyan	MediumOrchid	PaleGreen	SkyBlue	
BlanchedAlmond	DarkOliveGreen	GhostWhite	LightGoldenRodYellow	MediumPurple	PaleTurquoise	SlateBlue	
Blue	DarkOrchid	Gold	LightGray	MediumSeaGreen	PaleVioletRed	SlateGray	
BlueViolet	DarkRed	GoldenRod	LightGreen	MediumSlateBlue	PapayaWhip	SlateGrey	
Brown	DarkSalmon	Gray	LightGrey	MediumSpringGreen	PeachPuff	Snow	
BurlyWood	DarkSeaGreen	Green	LightPink	MediumTurquoise	Peru	SpringGreen	
CadetBlue	DarkSlateBlue	GreenYellow	LightSalmon	MediumVioletRed	Pink	SteelBlue	
Chartreuse	DarkSlateGray	Grey	LightSeaGreen	MidnightBlue	Plum	TECHNOLOGY	
Chocolate	DarkSlateGrey	HoneyDew	LightSkyBlue	MintCream	PowderBlue	Tan	
Coral	DarkTurquoise	HotPink	LightSlateGray	MistyRose	Purple	Teal	
CornflowerBlue	DarkViolet	IMPLEMENTATION	LightSlateGrey	Moccasin	Red	Thistle	
Cornsilk	Darkorange	IndianRed	LightSteelBlue	NavajoWhite	RosyBrown	Tomato	

Icons and Logos

```
@startuml
!include <logos/java>
!include <logos/ethereum>
!include <logos/ansible>

note as icons
| Category |= Expression |= Icon |
| Open Iconic | ""~<&align-left>"" | <&align-left
|   | ""~<&align-center>"" | <&align-center> |
|   | ""~<&align-right>"" | <&align-right> |
|   | ""~<&caret-left>"" | <&caret-left> |
|   | ""~<&caret-right>"" | <&caret-right> |
|   | ""~<&caret-top>"" | <&caret-top> |
|   | ""~<&caret-bottom>"" | <&caret-bottom> |
|   | ""~<&check>"" | <&check> |
|   | ""~<&circle-check>"" | <&circle-check> |
| SVG Logos | ""~<$java>"" | <$java> |
|   | ""~<$ethereum>"" | <$ethereum> |
|   | ""~<$ansible>"" | <$ansible> |
end note

@enduml
```



Category	**Expression**	**Icon**
Open Iconic	<&align-left>	≡
	<&align-center>	≡
	<&align-right>	≡
	<&caret-left>	◀
	<&caret-right>	▶
	<&caret-top>	▲
	<&caret-bottom>	▼
	<&check>	✔
	<&circle-check>	✅
SVG Logos	<$java>	
	<$ethereum>	
	<$ansible>	ANSIBLE

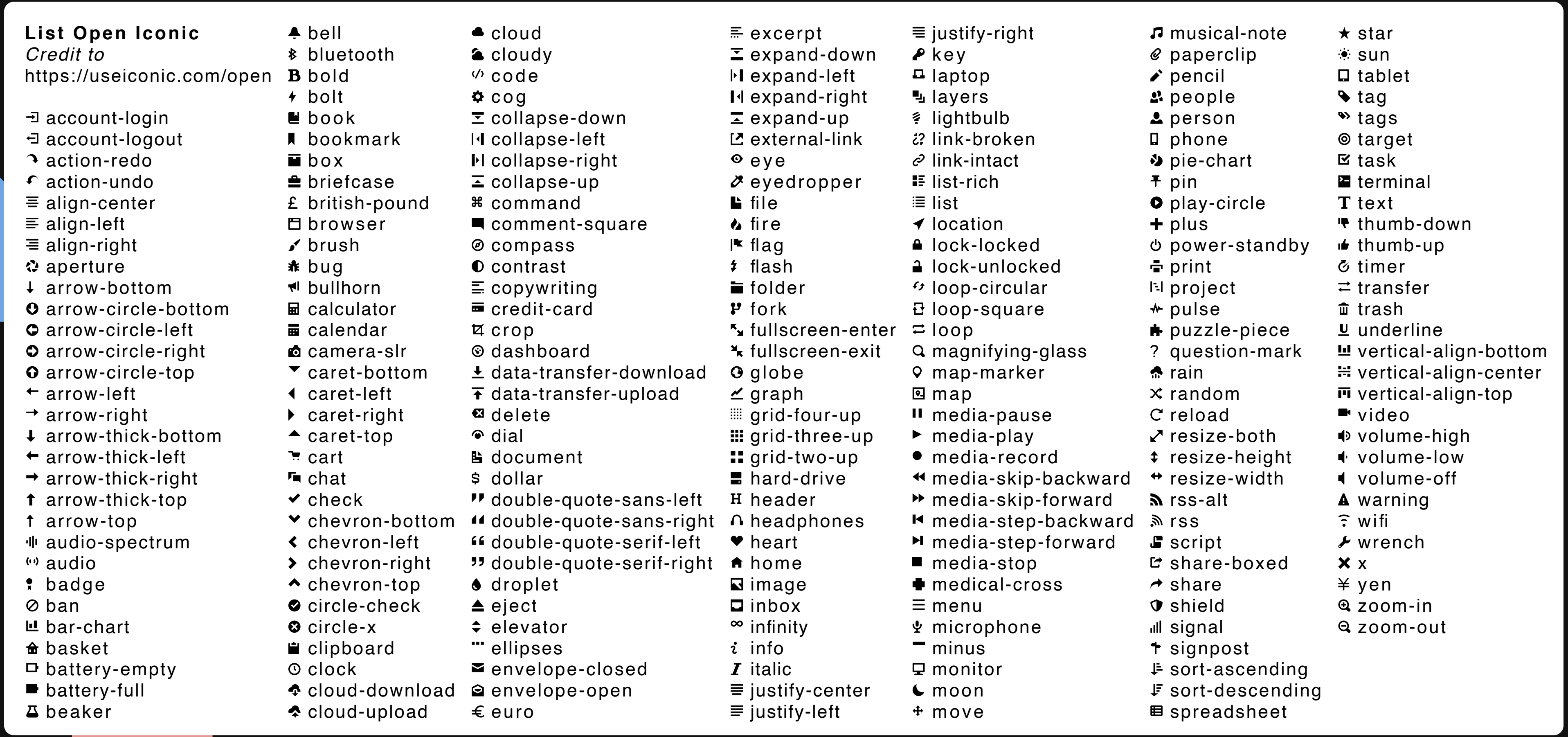
✅ Icon : `<&icon-name>`

✅ Logo : `<$logo-name>`

- References
  - PlantUML / Open Iconic
  - Open Iconic
  - PlantUML / Logos
  - SVG Logos

Icon List

```
@startuml
listopeniconic
@enduml
```



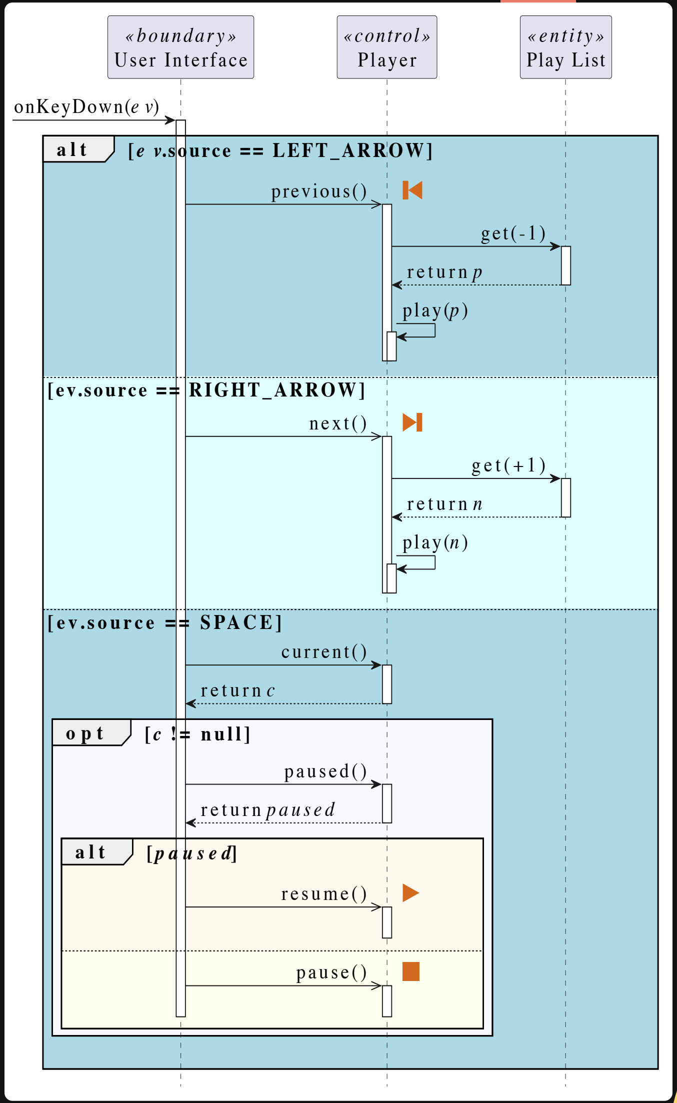
List Open Iconic
Credit to
https://useiconic.com/open

account-login
account-logout
action-redo
action-undo
align-center
align-left
align-right
aperture
arrow-bottom
arrow-circle-bottom
arrow-circle-left
arrow-circle-right
arrow-circle-top
arrow-left
arrow-right
arrow-thick-bottom
arrow-thick-left
arrow-thick-right
arrow-thick-top
arrow-top
audio-spectrum
audio
badge
ban
bar-chart
basket
battery-empty
battery-full
beaker
bell
bluetooth
bold
bolt
book
bookmark
box
briefcase
british-pound
browser
brush
bug
bullhorn
calculator
calendar
camera-slr
caret-bottom
caret-left
caret-right
caret-top
cart
chat
check
chevron-bottom
chevron-left
chevron-right
chevron-top
circle-check
circle-x
clipboard
clock
cloud-download
cloud-upload
cloud
cloudy
code
cog
collapse-down
collapse-left
collapse-right
collapse-up
command
comment-square
compass
contrast
copywriting
credit-card
crop
dashboard
data-transfer-download
data-transfer-upload
delete
dial
document
dollar
double-quote-sans-left
double-quote-sans-right
double-quote-serif-left
double-quote-serif-right
droplet
eject
elevator
ellipses
envelope-closed
envelope-open
euro
excerpt
expand-down
expand-left
expand-right
expand-up
external-link
eye
eyedropper
file
fire
flag
flash
folder
fork
fullscreen-enter
fullscreen-exit
globe
graph
grid-four-up
grid-three-up
grid-two-up
hard-drive
header
headphones
heart
home
image
inbox
infinity
info
italic
justify-center
justify-left
justify-right
key
laptop
layers
lightbulb
link-broken
link-intact
list-rich
list
location
lock-locked
lock-unlocked
loop-circular
loop-square
loop
magnifying-glass
map-marker
map
media-pause
media-play
media-record
media-skip-backward
media-skip-forward
media-step-backward
media-step-forward
media-stop
medical-cross
menu
microphone
minus
monitor
moon
move
musical-note
paperclip
pencil
people
person
phone
pie-chart
pin
play-circle
plus
power-standby
print
project
pulse
puzzle-piece
question-mark
rain
random
reload
resize-both
resize-height
resize-width
rss-alt
rss
script
share-boxed
share
shield
signal
signpost
sort-ascending
sort-descending
spreadsheet
star
sun
tablet
tag
tags
target
task
terminal
text
thumb-down
thumb-up
timer
transfer
trash
underline
vertical-align-bottom
vertical-align-center
vertical-align-top
video
volume-high
volume-low
volume-off
warning
wifi
wrench
x
yen
zoom-in
zoom-out

Sample Diagram 3

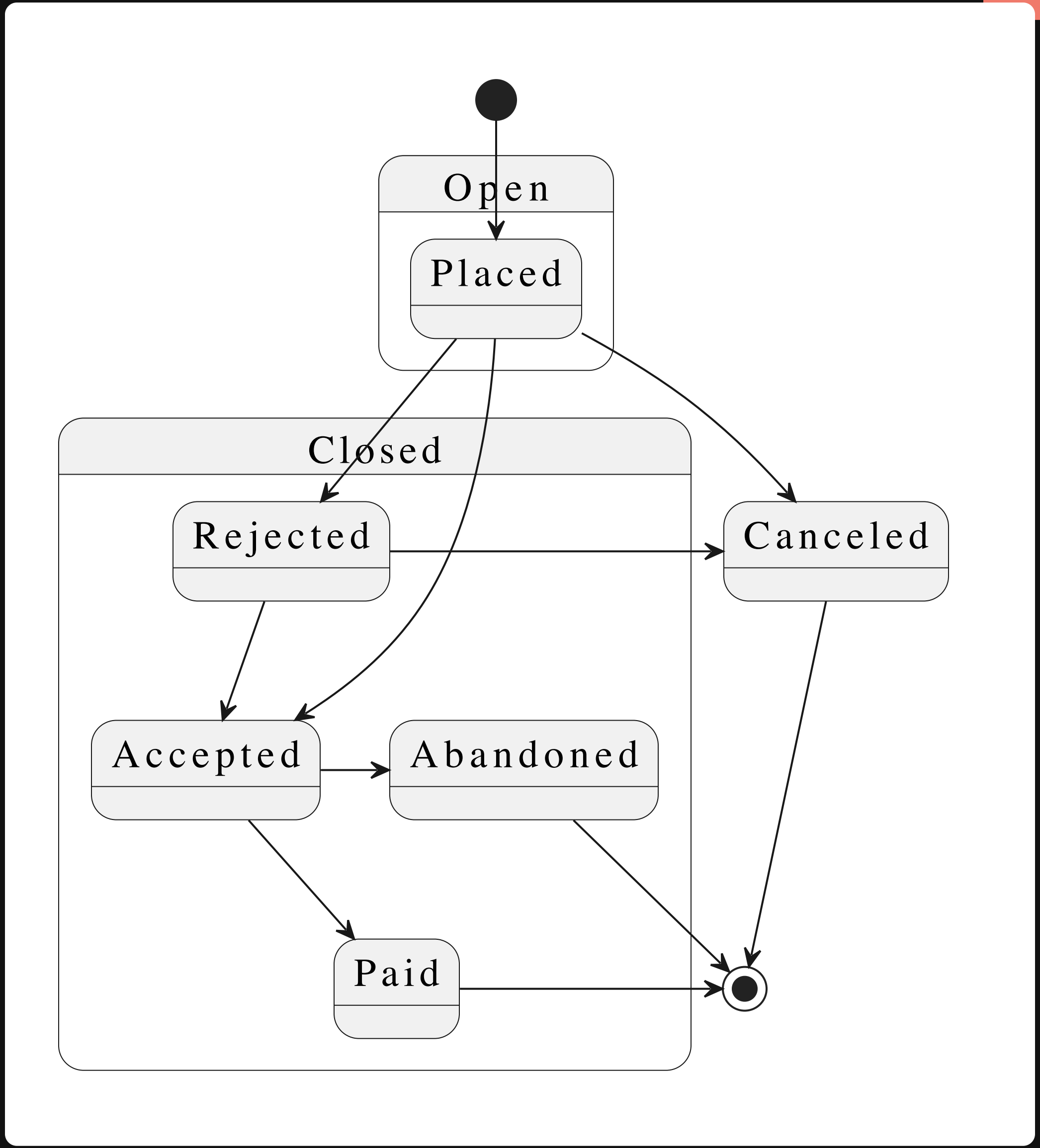
```
@startuml

'https://plantuml.com/sequence-diagram
skinparam DefaultFontName Consolas
skinparam DefaultFontSize 20
skinparam ParticipantPadding 40
skinparam SequenceMessageAlign reverseDirection
skinparam NoteBackgroundColor transparent
skinparam NoteBorderThickness 0
skinparam NoteFontSize 36
skinparam NoteFontColor chocolate

hide FootBox

participant "User Interface" as ui <<boundary>>
participant "Player" as player <<control>>
participant "Play List" as list <<entity>>

[-> ui ++ : onKeyDown(//ev//)
 alt #LightBlue //ev//.source == LEFT_ARROW
  ui ->> player ++ : previous()
   note right : <&media-step-backward>
   player -> list ++ : get(-1)
    return return //p//
   player -> player ++-- : play(//p//)
   player --
```

```
 else #LightCyan ev.source == RIGHT_
  ui ->> player ++ : next()
   note right : <&media-step-forwa
   player -> list ++ : get(+1)
    return return //n//
   player -> player ++-- : play(//
   player --
 else #lightBlue ev.source == SPACE
  ui -> player ++ : current()
   return return //c//
  opt #GhostWhite //c// != null
   ui -> player ++ : paused()
    return return //paused//
   alt #FloralWhite //paused//
    ui ->> player ++ : resume()
     note right : <&media-play>
     player --
   else #Ivory
    ui ->> player ++-- : pause()
     note right : <&media-stop>
     player --
   end
  end
  ui --
 |||
 end
@enduml
```



State Diagram - Sample

```
@startuml

state Open {
  state Placed
}
state Closed {
  state Rejected
  state Accepted
  state Abandoned
  state Paid
}

[*] --> Placed
Placed -right-> Canceled
Canceled -right-> [*]
Placed --> Rejected
Placed --> Accepted

Rejected -right-> Canceled
Rejected --> Accepted

Accepted --> Paid
Accepted -right--> Abandoned
Abandoned -right--> [*]
Paid -right--> [*]

@enduml
```



More on PlantUML

✓ Diagram Support

Category	Diagram
UML 2	Sequence Diagram
	State Diagram
	Activity Diagram
	Class Diagram
	Use-Case Diagram
Non-UML	Mind Map
	WBS
	Gantt Chart
	ERD

✓ Features

Feature	Description
Skinparams	Change colors and font of the drawing
Markup	A light-weight Creole engine is integrated in PlantUML
Colors	CSS named color and 6-digit RGB color are supported
Icons	Open Iconic is integrated in PlantUML
Logos	SVG Logos is suppored
Math	AsciiMath and LaTex are supported

✓ IDE Support

IDE	Plugin
VS Code	VS Code PlantUML Extension
IntelliJ IDEA	PlantUML Integration
Eclipse	PlantUML Eclipse Plugin