

# Dive into ICON - SCORE

ICON foundation

---

# Dive into ICON - SCORE

Step 1. Smart Contract - SCORE

Step 2. SCORE Guide

Step 3. SCORE Samples

Step 4. SCORE Development

---

# Dive into ICON - SCORE

## Step 1. Smart Contract - SCORE

1. What is SCORE
2. Feature of SCORE

## Step 2. SCORE Guide

## Step 3. SCORE Samples

## Step 4. SCORE Development

---

# Dive into ICON - SCORE

Step 1. Smart Contract - SCORE

Step 2. SCORE Guide

1. Token & Crowdsale
2. IconScoreBase abstract methods
3. ...

Step 3. SCORE Samples

Step 4. SCORE Development

---

# Dive into ICON - SCORE

Step 1. Smart Contract - SCORE

Step 2. SCORE Guide

Step 3. SCORE Samples

1. ICON Dice Roll
2. SampleGame (blackjack)
3. SCORE Style Guide

Step 4. SCORE Development

---

# Dive into ICON - SCORE

Step 1. Smart Contract - SCORE

Step 2. SCORE Guide

Step 3. SCORE Samples

Step 4. SCORE Development

1. Feature Requirements
2. Development & QnA

# 1. Smart Contract - SCORE

---

# Smart Contract - SCORE

- What is Smart Contract



# 1.1 What is SCORE

---

## Definition of SCORE

- SCORE in Dictionary : the number of points, goals, etc. achieved in a game or competition (from Cambridge Dictionary)
- ICON SCORE : Abbreviation of Smart Contract on Reliable Environment
- Definition of SCORE : Smart contract running on ICON network

## 1.2 Feature of SCORE

---

## Feature

- SCORE is written in python
- Uploaded as compressed binary data on the blockchain
- SCORE can be updated. SCORE address remains the same after update.
- SCORE code size is limited to about 64 KB (actually bounded by the maximum stepLimit value during its deploy transaction) after compression.
- SCORE must follow sandbox policy : file system access or network API calls are prohibited.

## 2. SCORE Guide

---

# SCORE Guide

- ICON Developers Portal
  - <https://www.icondev.io/docs/overview>
- iconservice API references
  - <https://icon-project.github.io/score-guide/api-references.html>

## 2.1 Token & Crowdsale

---

# Token & Crowdsale

- SCORE by example

<https://icon-project.github.io/score-guide/score-by-example.html#token-crowdsale>



## 2.2 IconScoreBase abstract methods

---

## IconScoreBase abstract methods

- IconScoreBase (The highest parent class)

<https://icon-project.github.io/score-guide/writing-score.html#iconscorebase-the-highest-parent-class>

- Abstract methods : `__init__`, `on_install`, `on_update`

## 2.3 DB abstraction

---

## DB abstraction

- VarDB, DictDB, ArrayDB

<https://icon-project.github.io/score-guide/writing-score.html#vardb-dictdb-arraydb>

- Available value type : **int**, **str**, **Address**, **bytes**

## 2.4 Decorator, fallback

---

## Decorator, fallback

- Decorator (external, eventlog, payable)

<https://icon-project.github.io/score-guide/writing-score.html#external-decorator-external>

- fallback

<https://icon-project.github.io/score-guide/writing-score.html#fallback>

## 2.5 Type hints, exception handling

---

# Type hints, exception handling

- Type hints

<https://icon-project.github.io/score-guide/writing-score.html#type-hints>

- Exception handling

<https://icon-project.github.io/score-guide/writing-score.html#exception-handling>



## 2.6 Global functions

---

# Global functions

- Global functions (json\_dumps, json\_loads, sha3\_256, revert)

<https://icon-project.github.io/score-guide/global-functions.html#global-functions>

## 2.7 InterfaceScore

---

# InterfaceScore

- InterfaceScore

<https://icon-project.github.io/score-guide/writing-score.html#interfacescore>

- Get InterfaceScore instance :

`create_interface_score('score address', 'interface class')`

## 2.8 Limitations

---

# Limitations

- Limitations

<https://icon-project.github.io/score-guide/limitation.html>

# 3. SCORE Samples

---

# SCORE Samples

- ICON Dice Roll
  - DApp (SCORE + GUI)
- SampleGame
  - Simple blackjack (Only SCORE)



## 3.1 ICON Dice Roll

---

# ICON Dice Roll

- Overview : Sample DApp using random generation. Supports single play
- SCORE source : GitHubGist

<https://gist.github.com/hx57/cc8a027a596e1e3676d59a6193d62c58#file-diceroll-py>

- DApp source : Medium Post

<https://medium.com/@2infiniti/icon-dapp-from-a-z-part-3-icon-dice-roll-dapp-7f0ca72057f5>

- Demo : <https://dapps.icon.support/icon-dice-roll/>

## 3.2 SampleGame

---

# SampleGame

- Overview : Sample SCORE implemented custom blackjack game. Supports Player vs Player game.
- SCORE source : GitHub repo

<https://github.com/icon-workshops/Dive-into-ICON-2-SCORE/tree/master/samplegame>

## 3.3 SCORE Style Guide

---

# SCORE Style Guide

- **ICON** specifies the **SCORE Style Guide** of components which **exposed to JSON-RPC**. (Should comply SCORE Style Guide prior to PEP 8)
- **Function** : camelCase
- **Parameters of function** : \_camelCase
- **SCORE params (on\_install)** : \_camelCase
- **Function with eventlog decorator** : PascalCase

# 4. SCORE Development

## 4.1 Feature Requirements



---

# Feature Requirements

- Initialize variables
- Save data (Transaction)
- Load data (Query)
- SCORE internal function call (InterfaceScore)
- SCORE integration test

## 4.2 Development & QnA

# Dive into ICON - Appendix

# Appendix A. Development Resources

---

# Development Resources

- GitHub
- Developer Portal
- ICON Improvement Proposal

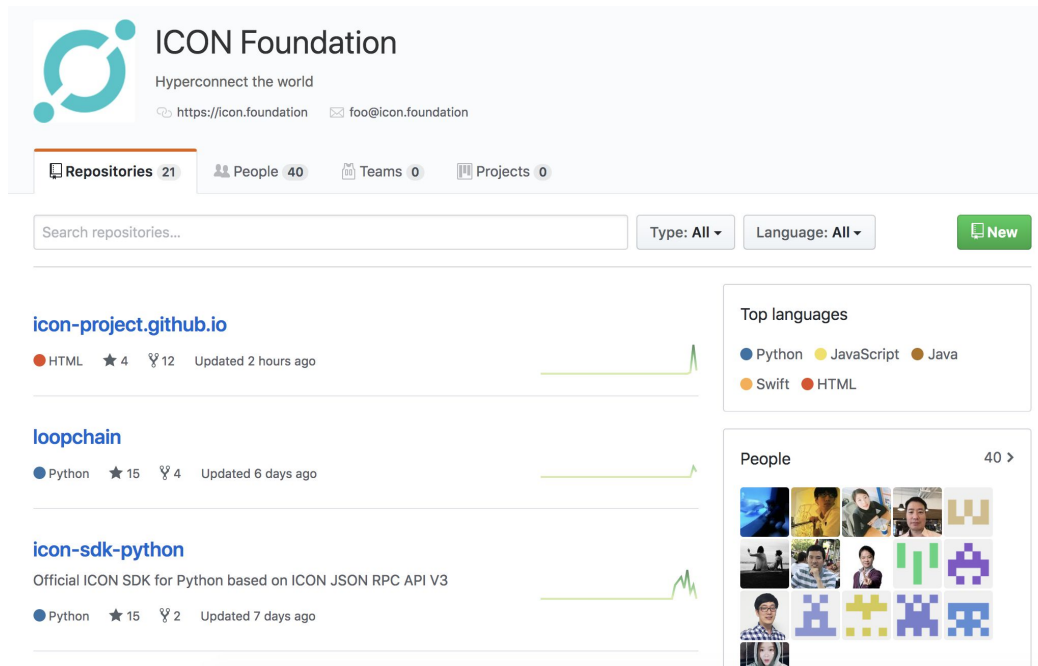
# GitHub <https://github.com/icon-project>

- loopchain
- icon-service
- icon-rpc-server

Node

- t-bears
- icon-sdk-python
- icon-sdk-java
- icon-sdk-js
- iconex\_android
- iconex\_ios
- iconex\_chrome\_extension

Dev tools



**ICON Foundation**  
Hyperconnect the world  
<https://icon.foundation> [foo@icon.foundation](mailto:foo@icon.foundation)

Repositories 21 People 40 Teams 0 Projects 0

Search repositories... Type: All Language: All New

**icon-project.github.io**  
HTML ★ 4 12 Updated 2 hours ago

**loopchain**  
Python ★ 15 4 Updated 6 days ago

**icon-sdk-python**  
Official ICON SDK for Python based on ICON JSON RPC API V3  
Python ★ 15 2 Updated 7 days ago

**Top languages**  
Python JavaScript Java Swift HTML

**People** 40 >

# Developer Portal <https://www.icondev.io>

- Community portal for ICON DApp ecosystem

## Getting Started

Tutorials for developers to get started

## SCORE

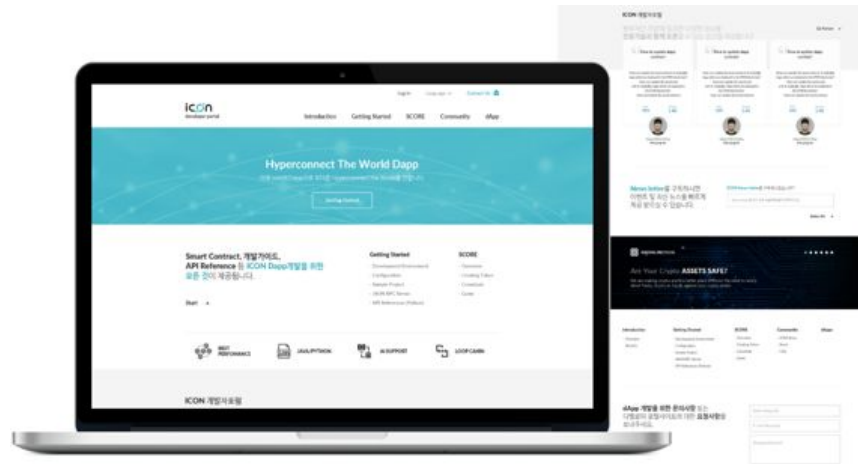
Details on ICON's Smart Contract, SCORE

## Community

Forum for Korean/English developers to discuss and communicate

## DApp

Overview of ICON DApp Partners



# ICON Improvement Proposal <https://github.com/icon-project/IIPs>

- IIP describes a standard for ICON platform.
- Anyone can prompt suggestions and discussions on new functions or improvement.
- Selected items will be implemented on ICON network.

- **For all other IIPs**, open a PR changing the state of your IIP to 'Final'. An editor will review your draft and ask if anyone objects to its being finalised. If the editor decides there is no rough consensus - for instance, because contributors point out significant issues with the IIP - they may close the PR and request that you fix the issues in the draft before trying again.

## IIP Status Terms

- **Draft** - an IIP that is open for consideration.
- **Last Call** - an IIP that is calling for last review before finalizing. IIPs that has been more than 2 weeks in Last Call without any technical changes or objections enters either Accepted or Final state.
- **Accepted** - an IIP that is planned for immediate adoption, i.e. expected to be included in the next release (for Core/Consensus layer IIPs only).
- **Final** - an IIP that has been adopted. For Core/Consensus layer IIPs, the implementation has been adopted in the mainnet.
- **Deferred** - an IIP that is not being considered for immediate adoption. May be reconsidered in the future.

## IIPs

Number	Title	Author	Type	Status
<a href="#">1</a>	IIP Purpose and Guidelines	Sojin Kim	Meta	Active
<a href="#">2</a>	ICON Token Standard	Jaechang Namgoong	IRC	Final
<a href="#">3</a>	ICON Non-Fungible Token Standard	Jaechang Namgoong	IRC	Draft
<a href="#">6</a>	ICON Name Service Standard	Phyrex Tsai, Portal Network Team	IRC	Draft



---

## Summary

- GitHub
- Developer Portal
- ICON Improvement Proposal