# kittyGame 智能合约api

/\*\*

### 创建猫的函数

@params address to 领养人地址

string genes 猫的基因

uint256 matronId & sireId 双亲id

uint32 generation 猫的代数

@return 返回uint256 猫的id

\*/

function createKitty(

address \_to,

string \_genes,

uint256 \_matronId,

uint256 \_sireId,

uint32 \_generation

) public returns(uint256)；

/\*\*

### 返回用户猫的数量

@params address owner 用户地址

@return uint 用户拥有数量

\*/

function balanceOf(address \_owner) public view returns(uint256)；

/\*\*

### 返回猫的拥有者

@params uint256 tokenId 猫的ID

@return address 猫的拥有者

\*/

function ownerOf(uint256 \_tokenId) public view returns(address) ；

/\*\*

### 获取用户拥有的猫的ID

@params address owner 用户地址

@return uint[] 用户拥有的猫的ID 数组

\*/

function getKittiesByOwner(address \_owner) external view returns(uint[])；

/\*\*

### 根据猫Id获取猫的详细数据

@params uint256 tokenId 猫的Id

@return bool isReady 是否在冷却

string genes 猫的基因

uint256 matronId & sireId 猫的双亲基因

uint256 birthTime 猫的出生时间

uint256 generation 猫的代数

\*/

function getKitty(uint256 \_tokenId) external view returns(

bool isReady,

string genes,

uint256 matronId,

uint256 sireId,

uint256 birthTime,

uint256 generation)；

/\*\*

### 创建售卖猫咪

@params uint256 tokenid 猫的Id

uint256 price 售卖的价格

\*/

function createAuction(uint256 \_tokenId, uint256 \_price) external onlyOwnerOf(\_tokenId)；

/\*\*

### 取消售卖猫

@params uint256 tokenId 猫的Id

\*/

function cancelAuction(uint256 \_tokenId) external onlyOwnerOf(\_tokenId)

/\*\*

### 获得猫的售卖信息

@params uint256 tokenId 猫的Id

@return address selle 出售者的地址

uint256 price 出售价格

\*/

function getAuctionByKitty(uint256 \_tokenId) public view returns (address seller, uint256 price)

/\*\*

### 判断猫是否在出售

@params uint256 tokenId 猫的Id

@return bool

\*/

function isSell(uint256 \_tokenId) public view returns(bool)

/\*\*

### 购买猫，同时需要支付以太币

@params uint256 tokenId 猫的Id

\*/

function buyKitty(uint256 \_tokenId) public payable

/\*\*

### 出售猫的交配权

@params uint256 tokenId 猫的Id

uint256 price 价格

\*/

function creatSireSell(uint256 \_tokenId, uint256 \_price) public onlyOwnerOf(\_tokenId)

/\*\*

### 取消出售猫的交配权

@params uint256 tokenId 猫的Id

\*/

function cancelSireSell(uint256 \_tokenId) public onlyOwnerOf(\_tokenId)

/\*\*

### 获取出售交配权的猫的详细信息

@params uint256 tokenId 猫id

@return address seller 出售交配权的猫的拥有着

uint256 price 交配权价格

\*/

function getSireByKitty(uint256 \_tokenId) public view returns(address seller, uint256 price)

/\*\*

### 判断猫是否出售交配权

@params uint256 tokenId 猫Id

@return boll

\*/

function isSireSell(uint256 \_tokenId) public view returns(bool)

/\*\*

### 购买猫的交配权，同时需要支付以太币

@params uint256 tokenId 猫Id

\*/

function buySire(uint256 \_tokenId) public payable

/\*\*

### 设置合约拥有者

@params address newOwner 合约新的主人

\*/

function setOwner(address \_newOwner) external onlyOwner