# 基本类及接口

class CCPlane :public Node { //飞机父类

public:

CREATE\_FUNC(CCPlane);

virtual bool init();

void setImg(const char \* filename);

void setAttri(int attack, int hp, int armor, int speed);

void changeAtk(int atk);

void changeHp(int hp);

void changeAr(int armor);

void changeSpd(int spd);

int getAtk();

int getHp();

int getAr();

int getSpd();

int getExp();

int getGrade();

void movePlane();

void atkPlane();

void skillPlane();

};

class Arms :public Node { //武器

public:

CREATE\_FUNC(Arms);

virtual bool init();

void setAtkValue(int atkValue);

void changeAtkValue(int changeValue); //成长值；

void setArmsImg(const char \* filename);

int getAtk();

};

class Flg : //机舱

public Node

{

public:

Flg();

~Flg();

CREATE\_FUNC(Flg);

virtual bool init();

void setAttri(int hp, int armor);

void changeAttri(int hp, int armor);

void setFlgImg(const char \* filename);

int getHp();

int getArmor();

class Fuel : //燃料

public Node

{

public:

Fuel();

~Fuel();

CREATE\_FUNC(Fuel);

virtual bool init();

void setSpeed();

void changeSpeed();

void setFuelImg(const char \* filename);

int getSpd();

｝

class FightLayer : //主游戏层

public Layer

{

public:

FightLayer();

~FightLayer();

CREATE\_FUNC(FightLayer);

virtual bool init();

void setPlayer(int cType);

virtual void onKeyPressed(cocos2d::EventKeyboard::KeyCode keyCode, cocos2d::Event\* event);

virtual void onKeyReleased(cocos2d::EventKeyboard::KeyCode keyCode, cocos2d::Event\* event);

virtual void getKey(float dt);

void bulletMove(float dt);

void addBullet(int bType);

void addEnemyPlane(int pType);

void is\_crash();

｝

class GameManager

{

public:

GameManager();

~GameManager();

static GameManager\* getInstance();

static void removeInstance();

void setPlane(CCPlane \* plane);

void setBullet(Bullet \* bullet);

void removePlane(CCPlane \* plane);

void removeBullet(Bullet \* bullet);

private:

static GameManager \* manager;

};

class Bullet:public Node

{

public:

Bullet();

~Bullet();

CREATE\_FUNC(Bullet);

virtual bool init();

void setBulletImg(const char \* filename);

void bulletMove(float dt);

void onEnterTransitionDidFinish();

void onExit();

private:

Sprite \* bulletImg;

};

class Support :public Node

{

public:

Support();

~Support();

void setSpt(int type);

void setSptImg(const char \* filename);

CREATE\_FUNC(Support);

virtual bool init();

int getType();

private:

Sprite \* sptImg;

int sTpye;

};