

# آموزش استفاده از سیستم ایجاد ذرات در GameMaker

بهترین اندازه دید : 768 \* 1024

نویسنده : مرتضی بصارت دار ( Mory )

[WWW.Persian-Designers.COM](http://WWW.Persian-Designers.COM)

کلیه حقوق این مقاله متعلق به نویسنده و سایت Persian Designers میباشد

بدون هیچ حرفی فایل آموزش را باز کرده و به قسمت create آبجکت ob\_fon بروید تا بحث را شروع کنیم .  
در نظر بیشتر افراد موتور ایجاد ذرات در گیم میکر بسیار پیچیده است . با این مثال می بینید که برعکس چقدر ساده و زیباست . یکی از نگرانی هایی که مانع از استفاده از موتور ذرات میشود این است که با این مشکلات روبه رو هستیم :

1- اینها چیستند و به چه درد میخورند ؟

- System ذرات
- Type ذرات
- نوع Emitter
- نوع destroyer

2- چگونه ذرات را در قسمت Step معرفی کرده تا شروع به حرکت کند .

3- چگونه سیستم ذرات را در قسمت Draw معرفی کنیم تا دیده شود .

در این مثال با یک نامگذاری مناسب تفاوت آنها را میبینید . این دستور را نادیده بگیرید در انتها توضیح داده میشود .

```
part_type_destroy_all();
```

در ابتدا باید یک سیستم ذرات را معرفی کنیم . این سیستم شامل مجموعه از متغیر ها میشود که در نهایت همه مشخصات ذرات را در خود خواهد داشت . در این مثال این سیستم favareh نامگذاری و بدین شکل معرفی میشود .

```
favareh= part_system_create();
```

اولین چیز معرفی یک قطره آب است . یعنی باید بگوییم ذره ای که باید ساخته شود آب است . پس نوع ذره را آب نامگذاری و بدین گونه معرفی میکنیم .

```
Aab = part_type_create();
```

حالا جزئیات را به آب اضافه میکنیم .

این دستورات به همراه چند دستور دیگر جزئیات دلخواه مان را به آب اضافه میکند .

اولین دستور میگوید آب به شکل يك پیکسل ساخته شود . ( در قسمت کدنویسی با نوشتن کلمه pt میتوانید انواع مختلفی از شکل ذره را در راهنمای برنامه ببینید . مثلا برای برف pt\_shape\_cross مناسبتر است . )

```
part_type_shape(aab,pt_shape_pixel);
```

اندازه ذرات . از کوچک به بزرگ . میزان رشد ذرات در هر step و اندازه اتفاقی رشد ذرات .

```
part_type_size(aab,10,10,1,1);
```

سرعت ذرات که بعدا در بخش Step معرفی میشود .

```
part_type_speed(aab,0,0,0,0);
```

نوع و تعداد ذراتی که باید در هر step تولید شود .

```
part_type_step(aab,4,pt_shape_square);
```

رنگی دست ساز برای تخصیص به آب .

```
myblue=make_color_rgb(104,184,242);
```

با این دستور رنگ شروع میان و پایان مسیر يك قطره تعیین میشود .

```
part_type_color(aab,myblue,myblue,c_white);
```

جاذبه . این اعداد را با توجه به سرعت اولیه آب تعیین کنید تا حرکات طبیعی به نظر برسد .

```
part_type_gravity(aab,-.5,-270);
```

تعداد Step هایی که يك قطره آب باقی میماند . در اینجا عددی اتفاقی بین 20 و 30 .

```
part_type_life(aab,20,30);
```

وضوح اولیه ، میانی و پایانی . قطره آب ابتدا کمتر دیده میشود سپس کاملا دیده میشود و در پایان راه خود تقریبا ناپدید میشود .

```
part_type_alpha(aab,.7,1,1);
```

مسیر پرتاب قطره ها . که در اینجا از 70 تا 110 درجه خواهد بود و 5 درجه هم به صورت اتفاقی کم و زیاد میشود .

```
part_type_direction(aab,70,110,0,5);
```

حالا تقریبا ویژگی های يك قطره آب در این سیستم معرفی شد . چیزهایی دیگر هم وجود دارد ولی فقط همین ها برای ما لازم بود . اما احتیاج به چیزی داریم که این آب ها از آن خارج شود . طبیعتا چیزی جز يك لوله آب نمیتواند باشد . Emitter آن چیزی است که در سیستم های ذرات این وظیفه را بر عهده دارد . پس ما هم يك Emitter را با نام لوله بدین شکل معرفی میکنیم .

اینجا گفتیم که در سیستم فواره يك Emitter با نام looleh ساخته شود .

```
looleh=part_emitter_create(favareh);
```

حال باید جزئیات لوله اضافه شود . این دستور میگوید لوله در سیستم favareh واز مختصاتی به فرم زیر و اشکال زیر باید قطره آب خارج کند . مختصات همان نوك لوله در شکل است .

```
part_emitter_region(favareh,looleh,x-1,x+2,y-30,y-32,ps_shape_rectangle,ps_distr_gaussian) ;
```

اما بالاخره این دستور که بسیار تعیین کننده است میگوید لوله در سیستم فواره باید 50 قطره آب در هر Step تولید کند . جمله واضحی است . نه ؟ هرچه میزان این عدد بیشتر باشد میزان آب بیشتری تولید میشود ولی حجم محاسبات سنگین تر میشود .

```
part_emitter_stream(favareh,looleh,aab,50);
```

حالا باید محدوده ای را معرفی کنیم که در آن قسمت ذرات از بین رفته و دیده نشود . اگر این کار را نکنید ذرات همه جا دیده میشوند و مشکلی ایجاد نمیشود . اما اینجا برای اینکه بدنه حوض وجود دارد . قسمت روبه روی حوض که دیده میشود به عنوان منطقه ممنوعه hoz به سیستم ذرات معرفی میشود . بدین شکل :

```
hoz=part_destroyer_create(favareh);
```

محدوده هم به صورت مستطیلی و به ابعاد زیر به سیستم معرفی شد .

```
part_destroyer_region(favareh,hoz,x-100,x+100,y-6,y+5,ps_shape_rectangle);
```

کارمان برای معرفی سیستم تمام شد .

اگر فایل را اجرا کنید می بینید که فشار آب کم و زیاد میشود . برای اینکار این متغیر ها را هم معرفی میکنیم . timer تنظیم کننده زمان تغییر سرعت یا فشار فواره است . spdnew سرعت هدف و spd سرعت جاری است . زیرا اگر بخواهیم سرعت را ناگهان تغییر دهیم غیر طبیعی خواهد بود . بعدا کار این متغیر ها را خواهید دید .

```
timer=0;  
spdnew=1;  
spd=3;
```

سرعت تکان خوردن آب شکل خواهد بود . ارتباطی به سیستم ندارد .

```
image_speed = 0.1;
```

## Step Event :

حالا وقت آن رسیده تا Step را تنظیم کنیم .

این دستور سرعت یا فشار آب را تنظیم میکند که از سرعت حد اقل و حداکثر و میزان افزایش را نشان میدهد . همیشه سعی کنید میزان حداقل و حداکثر متفاوت باشد زیرا شکلی طبیعی به آب میدهد .

```
part_type_speed(aab,spd,spd+1.5,,1,0);
```

این دستور میگوید ایجاد و تولید ذرات در سیستم يك گام پیش برود . مثلا اگر این دستور متوقف شود قطره ها ثابت در هوا خواهند ماند .

```
part_system_doastep(favareh);
```

این دستور می گوید اگر timer بزرگتر از 300 بود Alarm را فعال کن و timer را 0 کن . سپس يك واحد به تایمر اضافه کن . این دستور در مجموع هر 3 ثانیه يك بار آلارم را اجرا و باعث میشود سرعت آب تغییر کند .

```
if timer>300 {alarm[0]=10; timer=0;}  
timer+=1;
```

این دستور میگوید وقتی سرعت جاری از سرعت هدف کمتر بود کمی به سرعت جاری اضافه کن و به همین ترتیب سرعت را کم کن . همین دستور است که باعث میشود آب فواره به تدریج بالا و پایین رود . در ضمن سرعت هدف هم هر سه ثانیه به طور اتفاقی کم و زیاد میشود .

```
if spdnew>spd spd+=.05;  
if spdnew<spd spd-=.05;
```

## Draw Event :

حالا باید کار خود را به تصویر بکشیم . این دستور تصویر فواره را برایمان می کشد . دقت کنید اگر به جای 1- هر عددی باشد تصویر آب درون تصویر ثابت میماند .

```
draw_sprite(sprite_index,-1,x,y);
```

این دستور مورد نظر ما برای رسم سیستم ذرات favareh است . چون در قسمت لوله تعریف کردیم که آب از کجا خارج شود اینجا فقط نقطه صفر و صفر را نشان میدهیم .

```
part_system_draw(favareh,0,0);
```

کار تمام شده است . وقتی فایل را اجرا میکنید با فشردن دگمه هاي جهت بالا و پایین میتوانید سرعت فواره را کم و زیاد کنید .

اما :

مشکلی که در بازی نجات بندر به وجود آمد و بسیار هم جدی بود این بود که متاسفانه به دلیل نامعلومی مشخصات نوع ذرات با همان type با شروع مجدد از بین نمرود و باقی میماند . تعداد 3 عدد از يك نوع ذره کافیت که هر فایلی را سرنگون کند . برای پیشگیری از این مشکل از این دستور در ابتدای قسمت create استفاده شد .

```
part_type_destroy_all();
```

همین دستور باعث میشود تا قبل از شروع هر چیزی ابتدا تنظیمات گذشته ذرات پاک شود . این مشکل خطرناکی بود که اگر پیدا نمیشد به هیچوجه و در هیچ فایلی امکان استفاده از سیستم ذرات وجود نداشت .

کافیست با گذاشتن // در ابتدای این دستور آن را غیر فعال کرده و فایل را اجرا کنید و 3 بار با زدن Space ری استارت کنید .البته در همین رابطه 2 دستور دیگر نیز کاربرد دارند .

```
part_emitter_destroy_all(favareh);  
part_emitter_destroy(favareh,em);
```

ولی نبود هیچ کدام به اندازه نبود دستور اول مرگ آور نیست . و سوال اینجاست که چرا ؟

---

کلیه حقوق این مقاله برای نویسنده و سایت طراحان ایرانی ( [www.persian-designers.com](http://www.persian-designers.com) ) محفوظ می باشد  
استفاده از مطالب موجود در این مقاله در صورت ذکر کامل منبع مجاز است