|  |  |
| --- | --- |
| Make\_file(file\_name) | Maakt gwn een bestand aan. Deze functie word niet in het spel gebruikt |
| Players(world) | Deze functie is het beginscherm van het spel en vraagt de namen van de spelers. De functie returned de namen van de spelers zodra het ze allebij heeft en dan stopt het beginscherm ook. Het neemt world als een argument en world bevat bijna alle variabelen. |
| Search\_for(zoek, file) | Deze functie gaat op zoek naar een string in een bestand en returns wat er in de lijn eronder staat. Let wel het bestand mag geen lege spaties bevatten. Dit kan ik later nog aanpassen. |
| Class body | Dit is een classe die lichamen kan maken voor in het pygame spel te zetten. Het bevat vele nuttige functies die helpen met het maken en oproepen van bepaalde objecten. Het neemt ook vars uit de settings file zoals de breedte en hoogte als het een surface is. Of de size factor als het een image is. Bij het declareren moet je het een naam geven. Deze naam zal gebruikt worden bij het zoeken van var in het settings bestand. Het is als volgt opgesteld : naam + onderwerp bv: “gor1” + “\_hoogte” |
| Get\_info\_from(file) | Deze functie zit in body en in city. Het is belangrijk dat je deze functie gebruikt wanneer je een body maakt, anders zal de body niet weten hoe hij er uit moet zien en is hij dus leeg. Het zet ook de hoogt en breedte om naar ints |
| Pygamer(asked) in body | Deze functie maakt een surface of een rectangle aan voor een bepaald object |
| Get\_random\_x(screen, helft, file) | Deze funtcie bepaald een random x-coördinaat voor de gorillas. Je kan ook in settings intellen hoe ver ze minimaal van ekaar moeten staant. Je moet ook ingeven in welke helft van het scherm je de gorilla wil zetten |
| Latch(world) | Deze functie bekijkt in welk deel van het scherm de x-coördinaat van de gorilla ligt en bekijkt dan hoe hoog het gebouw is waar de gorilla op moet staan. Dan past het de y-coördinaat van de gorilla aan om op het gebouw te staan |

|  |  |
| --- | --- |
| Image(image\_file, file) | Als je een body hebt gemaakt en je wilt er een image van maken, dan doe je dat met image. Het neemt maakt een surface van je image en dan haalt het de size\_factor uit de settings. Dit zal de image groter of kleiner maken. Het past de breedte en hoogte van je body aan en het maakt een surface en een rectangle |
| Make\_hitbox(file) | Als je een body die je hebt gemaakt een hitbox wilt geven doe je dat best met de functie make\_hitbox. Je kan ook de rectangle van je body als hitbox gebruiken maar op die manier kan je de grote niet aanpassen. Deze functie maakt een hitbox aan op de coördinaten van je body. Let wel op deze hitbox hangt niet vast op je body dus als de body beweegt zal de hitbox blijven staan. De hitbox kan je ook weer groter of kleiner maken in de settings |
| Class city | Dit is een aparte classe speciaal aangemaakt om de gebouwen op het spel te zetten. Zodra je city declareert zal deze classe direct gebouwen aanmaken met een for loop. De grote van deze gebouwen moet ook weer aangegeven zijn in de settings. De afmetingen van de buildings worden opgeslagen in een list en de lengte van de list word opgeslagen in n\_buildings. |
| Pygamer(asked) in city | Doet hetzelfde als in body, maar nu kan je ook de hoogte van de gebouwen vragen. |
| Class map | Deze classe bevat alle belangrijke variabelen. Dit is gwn zodat ik niet 10 argumenten in één functie moet vragen. |
| Render\_buildings(buildings, screen) | Zet alle buildings op het scherm. Let wel buildings moet een object zijn van city |
| Ask\_angle(screen, breedte, player) | Deze functie vraagt de hoek waarin de speler wilt gooien. Het geeft een lijst terug met daarin [continue, angle]. De speler word dus gevraagd een hoek in te geven, zodra hij de hoek ingeeft en op spatie drukt, zal continue True worden. |
|  |  |