» arukas_container

Provides container resource. This allows container to be created, updated and deleted.

For additional details please refer to API documentation.

» Example Usage

Create a new container using the "NGINX" image.

```
resource "arukas_container" "foobar" {
  name      = "terraform_for_arukas_test_foobar"
  image     = "nginx:latest"
  instances = 1
  memory    = 256

  ports = {
    protocol = "tcp"
    number   = "80"
  }

  environments {
    key   = "key1"
    value = "value1"
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required, string) The name of the container.
- `image` - (Required, string) The ID of the image to back this container.It must be a public image on DockerHub.
- `instances` - (Optional, int) The count of the instance. It must be between `1` and `10`.
- `memory` - (Optional, int) The size of the instance RAM.It must be `256` or `512`.
- `endpoint` - (Optional,string) The subdomain part of the endpoint assigned by Arukas. If it is not set, Arukas will do automatic assignment.
- `ports` - (Required , block) See Ports below for details.
- `environments` - (Required , block) See Environments below for details.
- `cmd` - (Optional , string) The command of the container.

» Ports

`ports` is a block within the configuration that can be repeated to specify the port mappings of the container. Each `ports` block supports the following:

- `protocol` - (Optional, string) Protocol that can be used over this port, defaults to `tcp`,It must be `tcp` or `udp`.
- `number` - (Optional, int) Port within the container,defaults to `80`, It must be between `1` to `65535`.

» Environments

`environments` is a block within the configuration that can be repeated to specify the environment variables. Each `environments` block supports the following:

- `key` - (Required, string) Key of environment variable.
- `value` - (Required, string) Value of environment variable.

» Attributes Reference

The following attributes are exported:

- `id` - The ID of the container.
- `app_id` - The ID of the Arukas application to which the container belongs.
- `name` - The name of the container.
- `image` - The ID of the image to back this container.
- `instances` - The count of the instance.
- `memory` - The size of the instance RAM.
- `endpoint` - The subdomain part of the endpoint assigned by Arukas.
- `ports` - See Ports below for details.
- `environments` - See Environments below for details.
- `cmd` - The command of the container.
- `port_mappings` - See PortMappings below for details.
- `endpoint_full_url` - The URL of endpoint.
- `endpoint_full_hostname` - The Hostname of endpoint.

» PortMappings

`port_mappings` is a block within the configuration that the port mappings of the container. Each `port_mappings` block supports the following:

- `host` - The name of the host actually running the container.
- `ipaddress` - The IP address of the host actually running the container.
- `container_port` - Port within the container.
- `service_port` - The actual port mapped to the port in the container.