Runtime JavaScript Code

This directory contains Deno runtime code written in plain JavaScript.

Each file is a plain, old **script**, not ES modules. The reason is that snapshotting ES modules is much harder, especially if one needs to manipulate global scope (like in case of Deno).

Each file is prefixed with a number, telling in which order scripts should be loaded into V8 isolate. This is temporary solution and we're striving not to require specific order (though it's not 100% obvious if that's feasible).

Deno Web APIs

This directory facilities Web APIs that are available in Deno.

Please note, that some implementations might not be completely aligned with specification.

Some Web APIs are using ops under the hood, eg. `console`, `performance`.

Implemented Web APIs

- Blob: for representing opaque binary data.
- Console: for logging purposes.
- CustomEvent, EventTarget and EventListener: to work with DOM events.
  - **Implementation notes:** There is no DOM hierarchy in Deno, so there is no tree for Events to bubble/capture through.
- fetch, Request, Response, Body and Headers: modern Promise-based HTTP Request API.
- location and Location.
- FormData: access to a `multipart/form-data` serialization.
- Performance: retrieving current time with a high precision.
- setTimeout, setInterval, clearTimeout: scheduling callbacks in future and clearInterval.
- Stream for creating, composing, and consuming streams of data.
- URL and URLSearchParams: to construct and parse URLSs.
- Worker: executing additional code in a separate thread.
  - **Implementation notes:** Blob URLs are not supported, object ownership cannot be transferred, posted data is serialized to JSON instead of structured cloning.