



A_DATEN_CODIEREN_TXT_ANNEX

Aufgaben

1. Besorgen sie sich im Internet eine vollständige ASCII-Tabelle als Bild- oder PDF-Datei und ergänzen sie damit ihr ePortfolio.
2. Suchen sie im Internet nach einer Webseite, wo Unicode-Zeichen gelistet sind und merken bzw. notieren sie sich den Link.
3. Sie erhalten eine ZIP-Datei Textsamples.zip unter folgendem Link:
<https://juergarnold.ch/Codesysteme/Textsamples.zip>
Laden sie die ZIP-Datei auf ihren Notebook und extrahieren sie die drei Dateien **Textsample1**, **Textsample2** und **Textsample3**. Eine der drei Dateien ist in ASCII codiert, die andere in UTF-8 und die dritte in UTF-16. Beantworten sie nun die folgenden Fragen:
 - a. Welche der Dateien ist nun ASCII-codiert, welche UTF-8 und welche UTF-16 BE-BOM?
 - b. Alle drei Dateien enthalten denselben Text. Aus wie vielen Zeichen besteht dieser?
 - c. Was sind die jeweiligen Dateigrößen? (Beachten sie, dass unter Grösse auf Datenträger jeweils 0 Bytes angegeben wird. Dies darum, weil beim Windows-Dateisystem NTFS kleine Dateien direkt in die MFT (Master File Table) geschrieben werden.) Wie erklären sie sich die Unterschiede?
 - d. Bei den weiteren Fragen interessieren uns nur noch die ASCII- und die UTF-8-Datei: Bekanntlich ist UTF-8 in den ersten 128 Zeichen deckungsgleich mit ASCII. Untersuchen sie nun die beiden HEX-Dumps und geben sie an, welche Zeichen unterschiedlich codiert sind. Ein kleiner Tipp: Es sind deren zwei.
 - e. Was bedeuten die beiden Ausdrücke, denen wir z.B. bei UTF-16 begegnen: Big-Endian (BE), Little-Endian (LE)?
 - f. Im Notepad++ kann man unter dem Menüpunkt Codierung von ASCII zu UTF umschalten. Spielen sie damit etwas herum und notieren sie sich, was in der Darstellung jeweils ändert.
 - g. Für Anspruchsvolle: Der UTF-8-Code kann je nach Zeichen ein, zwei, drei oder vier Byte lang sein. Wie kann der Textreader erkennen, wann ein UTF-8 Zeichencode beginnt und wann er endet? Untersuchen sie dies anhand der beiden Textsamples und lesen sie in z.B. Wikipedia die entsprechende Theorie zu UTF-8 durch. Tipp: Startbyte und Folgebyte.



4. Einfacher QR-Code erstellen und lesen:

Denken sie sich eine kurze Botschaft, URL etc. aus und bilden sie diese Information in einem QR-Code ab. Danach tauschen sie mit ihrem Banknachbar/in ihre QR-Codes aus. Wenn der QR-Code gelesen werden kann, waren sie erfolgreich. Applikationen die alphanumerischen Text in einen QR-Code und zurück wandeln, findet man im Internet.

5. Zusammengesetzte Daten in einem QR-Code:

Ihre Firma wird mit dem Design von Tickets für ein Fussballstadion beauftragt. Das Eintrittsticket soll mit einem QR-Code versehen sein, der alle wichtigen Informationen zur Buchung enthält. Dies soll ermöglichen, das Ticket jederzeit und überall von Offline-QR-Readern lesen und überprüfen zu lassen bzw. Zugang zu den Stadionbereichen zu gewähren. Das Ticket soll die folgenden, codierten Informationen enthalten:

- a. Datum und Uhrzeit der Veranstaltung
 - b. Fortlaufende, alphanumerische Ticketnummer (Ticketnummern seit Tag-0)
 - c. Numerische Sitzplatznummer (Jeder Sitzplatz hat eigene Nummer)
 - d. Tribünensektor A-Z (Für einfache Platzeinweisung der Besucher)
 - e. ID- oder Passnummer des Besuchers (Tickets nicht übertragbar)
- Überlegen sie sich, ob das Ticket fälschungssicher ist. Braucht es allenfalls noch eine Prüfziffer, Checksum etc.?
 - Wenn der Besucher am Stadioneingang erscheint, zeigt er sein Ticket. Dieses wird von einem Platzanweiser mit einem QR-Code-Leser gelesen. Damit kann er die Personalie und Gültigkeit des Tickets überprüfen und den Besucher anschliessend in den richtigen Stadionsektor leiten.
 - Bilden sie diesen Datensatz in einen QR-Code ab. Erstellen sie QR-Codes für fiktive Veranstaltungen. Testen sie ihre QR-Codes mit ihrem Banknachbarn/in gegenseitig aus.
 - Applikationen die alphanumerischen Text in einen QR-Code und zurück wandeln, findet man im Internet.
 - Auf Komplettlösungen aus dem Internet - es gibt zu diesem Thema mehr oder weniger "pfannenfertige" Applikationen - bitte aber verzichten.